Title of the thesis	**Parameter-free agglomerative hierarchical clustering to model learners' activity in online discussion forums**
Author	**Germán Cobo Rodríguez** IT, Multimedia and Telecommunications Department Universitat Oberta de Catalunya (UOC)
Co-directors	**Dr. Eugènia Santamaría Pérez** **Dr. Jose Antonio Morán Moreno** IT, Multimedia and Telecommunications Department Universitat Oberta de Catalunya (UOC)
Adviser	**Dr. Joan Claudi Socoró Carrié** Communications and Signal Theory Department Escola Tècnica Superior d'Enginyeria Electrònica i Informàtica La Salle
Doctoral programme	**Information and Knowledge Society** Internet Interdisciplinary Institute (IN3) Universitat Oberta de Catalunya (UOC)
Deposit date	**January 20, 2014**

This document was typeset by the author using LaTeX.

The author confirms that the work submitted is his own and that the appropriate credit has been given where references have been made to the work of others.

"People assume that time is a strict progression of cause to effect, but actually, from a non-linear non-subjective viewpoint, it's more like a big ball of wibbly-wobbly... timey-wimey... stuff."

The Doctor

Acknowledgements

Looking for plenty of proper nouns? Wrong timeline, sweetie. Perhaps expecting some pleasant, sugar-coated, devoted lines? Erroneous version of reality, dear. If that's your poison, I recommend you to omit this page. Just for the sake of maximising your own time.

Those, and they are many, who I am honestly grateful to don't need any inscription here to be aware of my gratefulness. If that's not your case, no offence, but maybe you should consider the possibility of not being one of them. And that's essentially all I have to say on the matter.

The only liberty I'm going to take here is to encourage you, whoever you are, to read an Alessandro Baricco's very funny novel titled "City". Apart from being really great literature, this novel hides a priceless pearl: Professor Mondrian Kilroy's "Essay on Intellectual Honesty". I firmly believe that everyone should read, enjoy and reflect on it, specially those gentle spirits who are about to become PhD students.

And, like a man of huge talent and wisdom once wrote, "that's the way it crumbles, cookie-wise", and otherwise-wise.

Abstract

The analysis of learners' activity in online discussion forums leads to a highly context-dependent modelling problem, which can be posed from both theoretical and empirical approaches. When this problem is tackled from the data mining field, a clustering-based perspective is usually adopted, thus giving rise to a clustering scenario where the real number of clusters is *a priori* unknown. Hence, this approach reveals an underlying problem, which is one of the best-known issues of the clustering paradigm: the estimation of the number of clusters, habitually selected by user according to some kind of subjective criterion that may easily lead to the appearance of undesired biases in the obtained models.

With the aim of avoiding any user intervention in the cluster analysis stage, two new cluster merging criteria are proposed in the present thesis, which allow to implement a novel parameter-free agglomerative hierarchical clustering algorithm. A complete set of experiments indicates that the new clustering algorithm is able to provide optimal clustering solutions in the face of a great variety of clustering scenarios, both having the ability to deal with different kinds of data and outperforming clustering algorithms most widely used in practice.

Finally, a two-stage analysis strategy based on the subspace clustering paradigm is proposed to properly tackle the issue of modelling learners' participation in the asynchronous discussions. In combination with the new clustering algorithm, the proposed strategy proves to be able to limit user's subjective intervention to the interpretation stages of the analysis process and to lead to a complete modelling of the activity performed by learners in online discussion forums.

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Framework of the thesis

The original motivation that gives rise to the present thesis is found in the analysis of learners' activity in online discussion forums, which are one of the most common tools in online teaching-learning environments. In addition to both the studies set out from different theoretical perspectives and the approaches based on social network analysis, the issue of modelling learners' participation in discussion forums is posed from the data mining field as a clustering scenario, where learners with similar activity patterns are grouped together and the analysis of the resultant clusters leads to the identification of different learning behaviours. Since the participation patterns that can arise from asynchronous discussions in online forums depend on many variables (*e.g.* amount of learners in the virtual classroom, teaching-learning strategies promoted by teacher, kind of subject, etc.), the real number of clusters in such scenario is *a priori* unknown. Hence, this approach reveals an underlying problem, which is one of the best-known issues of the clustering paradigm: the estimation of the number of clusters. Despite the fact that, like in many other applications of clustering algorithms, the final set of clusters can be manually selected under some kind of subjective criterion, an automatic estimation based only on the data itself would be a better solution in order to successfully find the real number of clusters. Thus, the main goal of this thesis is focused on the issue of the automatic estimation of the number of clusters in clustering problems and its application to the analysis of learners' activity in online discussion forums.

In this first chapter, the global framework of the thesis is defined. Firstly, the presence of online discussion forums in the context of the online learning environments is described in section 1.1. Next, in section 1.2, different approaches to the issue of modelling learners' activity in online discussion forums are presented, including a general conceptual frame (section 1.2.1) and different perspectives on modelling strategies (sections 1.2.2 and 1.2.3). Finally, both research questions and hypotheses, as well as the structure of the present thesis, are defined in section 1.3.

1.1 Discussion forums in the online learning context

The term online learning (or e-learning) usually refers to any transference of skills and knowledge where the information and communication systems serve as specific media to implement the teaching-learning process (Tavangarian et al., 2004). Although the great diversity of terminologies and definitions used for online learning that exist in literature have been formulated from many different standpoints (Keegan, 1990; Khan, 1997; Relan and Gillani, 1997; Carliner, 1999; Cole, 2000; Rosenberg, 2001; Rossett, 2002; Garrison and Anderson, 2003; Tavangarian et al., 2004; Ally, 2008), one of their most important common factors is the concept of interaction (Vrasidas, 1999). In this sense, interaction reveals itself as one of the main components not only in the online learning context, but in any teaching-learning experience (Vygotsky, 1978).

Interaction, in all of its forms, has been identified as one of the main topics in distance education research (Wagner, 1994; Gunawardena and McIsaac, 2004). In fact, a high level of interaction is usually desirable and increases the effectiveness of distance education courses (Fulford and Zhang, 1993). According to Moore (1989), three different types of interaction may be considered: learner-content interaction, which refers to the relation between learners and all the contents, information and ideas they encounter in their course materials; learner-instructor interaction, which provides advice, clarification and feedback between learners and teachers; and learner-learner interaction, which allows learners to dialogue among themselves in order to exchange opinions, thoughts and ideas. Pointing out the fact that, for any type of interaction to take place in a distance education context, learners have to interact with the medium, a fourth type of interaction is proposed by Hillman et al. (1994): learner-interface interaction, which defines interface not as a mediating element in all interaction, but as an active and independent mode of interaction that has an impact over the learning experience and with which learners have to deal. In this way, the interaction processes that occur in an online teaching-learning environment, as well as their derived consequences, are both defined and influenced by –among other factors such as course structure, provided feedback and class size– those mediated communication forms that make them possible (Vrasidas and McIsaac, 1999).

In general terms, computer-mediated communication can be defined as a process of communication via computers (or networked computers, or the Internet), involving people, situated in particular contexts, engaging in processes to shape media for a variety of purposes (December, 1996). From an educational perspective, computer-mediated communication has been principally understood as a facilitator of critical thinking, collaborative learning and knowledge building (Harasim, 1989), as an effective and beneficial pedagogical asset (Althaus, 1997) and as a proper support for conversational models of learning (Laurillard, 1999). As online teaching-learning environments have become more complex and sophisticated, computer-mediated communication has proved to have great potential for designing learning tools that serve as a support and media-

tion for communication and interaction among learners and teachers. Furthermore, it has become established as one of the most influential features in the context of online learning environments (Yukselturk, 2010).

According to whether they require temporal concurrence of users or not, computer-mediated communication tools are generally classified as synchronous or asynchronous (Hines and Pearl, 2004). On the one hand, synchronous communication tools give rise to teaching-learning processes that occur in real time and need for the simultaneous participation of both learners and teachers (Romiszowski and Mason, 2004). Despite some identified limitations, such as the necessity of getting learners and teachers online at the same time, difficulty in moderating larger-scale conversations, lack of reflection time for learners or intimidation of poor typists, synchronous tools like text chat, audio/video-conferencing and virtual whiteboards, among others, have proved to be useful for collaborative decision-making, brainstorming, community building and dealing with technical issues (Branon and Essex, 2001). On the other hand, teaching-learning processes that take place by means of asynchronous communication tools occur in delayed time and do not require the simultaneous participation of both learners and teachers (Johnson, 2006). The most common lacks associated to asynchronous tools like email and discussion forums are a possible absence of immediate feedback, large lengths of time for discussions to mature and a possible sense of isolation felt by learners. In contrast, they have many and well-known advantages, such as encouraging in-depth and more thoughtful discussions, communicating with learners and teachers under no time constraints, holding ongoing discussions where archiving is required and allowing all learners to respond to a topic (Branon and Essex, 2001).

Both synchronous and asynchronous ways of communication are necessary for online learning (Davidson-Shivers et al., 2001). Nonetheless, since they facilitate more unconstrained discussions of any kind (Sun et al., 2011), asynchronous communication tools are more often used in virtual learning environments than the synchronous ones, which are frequently considered as optional course features and used for more specific purposes (Burnett, 2003). In fact, more concretely, the literature in this area indicates that online discussion forums –also known as discussion boards, bulletin boards, threaded discussions or message boards– are one of the most widely spread and primarily used communication tools in educational contexts. Bauer (2002) observes, in his study on how learners can be assessed from the discussion forums, that almost every online course web site contains a discussion board, where class members can post messages, exchange ideas and ask questions. McLoughlin (2002) finds, in a study of undergraduate teams working online to complete tasks, that the majority of teams, including the most successful ones, actively used the forums to share ideas and discuss the specifics of their projects. In the same way, the study carried out by Paulus (2007), where the learning processes performed by different working groups are decomposed in a succession of steps of diverse nature (*e.g.* check-in, brainstorm ideas, assign tasks, combine contributions, provide and integrate feedback, etc.), show as well that discussion forums are by far the communication tool most widely used by all groups throughout the majority of steps.

Therefore, due to their popularity and demonstrated effectiveness, discussion forums clearly play an important role in online teaching-learning environments (Rourke and Anderson, 2002; Wu and Hiltz, 2004).

1.1.1 Online discussion forums from learner and teacher perspectives

Generically, an online discussion forum has been defined as a tool that allows and facilitates asynchronous, computer-mediated, multi-directional, text-based and topic-threaded online communication (Yang, 2004). From an educational point of view, extensive studies that can be found in literature have been developed on how to use online discussion forums as a teaching and learning tool (Collins, 1998; Branon and Essex, 2001; Land and Dornisch, 2002). Online discussion boards have been pointed to as a useful tool, both for teachers and learners, in order to develop strategies for building collaborative problem-solving courses and designing discovery-oriented activities (Scardamalia and Bereiter, 1996), as well as for giving and accepting feedback and for greater reflection (MacKnight, 2000). Also, they have been viewed to have a great potential as an assessment tool (McLoughlin and Luca, 2001). Moreover, it has been argued that interaction through online discussion forums promotes student-centred learning, encourages wider learner participation, produces in-depth and reasoned discussions (Karayan and Crowe, 1997; Smith and Hardaker, 2000a), is remarkably task-oriented and reflects high phases in knowledge construction (Aulls et al., 2010).

From a learner's perspective, asynchronous learning through online discussion forums allows working with no time constraints, since they can access the online materials and interact with other learners and teachers at anytime (Ally, 2008). Since discussion, or dialogue, is a valuable educational tool, that helps in students' learning (Larson, 2000; Laurillard, 2002; Winiecki, 2003) and enhances the learning process by creating more opportunities for active learning and collaboration (Klemm, 1997; Landsberger, 2001; Land and Dornisch, 2002), conferencing and interchanging of ideas and knowledge through online discussion boards have become a basic component of student-centred web-based courses and a main vehicle for contact between learners (Rossman, 1999; Brown, 2001).

Although they lack the immediacy of live communication, discussion boards lead students up to a more directed and lasting flow of concepts, ideas and opinions (McCampbell, 2000). When using online discussion forums, learners are required to post questions, answers and ideas, read and respond to other learners' contributions and post new messages to clarify or revise their opinions. These activities involve students into a series of complex cognitive procedures, such as formulating ideas into words, evaluating the viewpoints of others, negotiating meanings with teachers and other students, and modifying their original ideas. In this way, discussion through online forums is beneficial for both student involvement and learning outcomes (Harris and Sandor, 2007) and, from a social-constructivist point of view, helps to build a learning community (Cooper, 2000;

Tiene, 2000; Brown, 2001) and facilitates learners to construct knowledge (Rovai, 2000; Campos et al., 2001; Gray, 2002; Laurillard, 2002).

From a teacher's perspective, online discussion forums are a popular and very useful tool that, due to its asynchronous nature, allows them to perform their tutoring tasks at anytime and from anywhere (Bauer, 2002). The role of the teacher in managing discussion boards in a virtual classroom has been described in many different ways, such as moderator, facilitator, helper, guide and role model (Landsberger, 2001; Muirhead, 2002; Wozniak and Silveira, 2004). Discussion boards provide teachers with an alternative method both to interact with learners (Yang, 2004) and to increase interactivity among them (Klemm, 1997; Bannan-Ritland, 2002). Through online discussion forums, teachers can observe learners' reaction to instruction and monitor their knowledge construction process (Jones and Harmon, 2002), as well as they can analyse the discussions content in order to give feedback (MacKnight, 2000) and assess learners' higher order activities, critical thinking and problem solving skills (McLoughlin and Panko, 2002).

In fact, part of the gap between learning theory (descriptions of how learners learn) and instructional theory (prescriptions for teachers to design and give courses) can be bridged through the analysis and modelling of learners' activity in asynchronous discussions (Knowlton, 2005). According to many authors, these analysis and modelling procedures are necessary for teachers and instructors to properly execute some of their most laborious tasks in the context of the online learning discussions, such as trying to move passive students to more active types of participation (Salmon, 2000), changing the mindset of learners to help them break out of their stereotypical roles of information receivers and take the roles of seekers, explorers and users (Prestera and Moller, 2001), and, in general terms, facilitating learners' interactions beyond not simply playing the expected role of stimulus providers (Morrison and Guenther, 2000). Furthermore, learners' needs, skills and level of expertise can be determined by analysing their participation and patterns of interaction throughout the online threaded discussions (Winiecki, 2003), so that teachers can also use this information to develop and adapt teaching-learning strategies in order to enhance students' learning outcomes (Ally, 2008).

However, since courses enrolments can easily reach several hundreds of learners, the analysis of the resultant online interaction that take place in discussion boards can become a considerable burden on teachers (Kim et al., 2011). Thus, there exist, from teacher's point of view, both the interest in and the need for having instructional tools that allow studying and modelling the activity and the interaction patterns performed by learners in online discussion forums (Thomas, 2002). Inasmuch as the way students participate in online discussion boards can be a very useful source of indicators for teachers in order to facilitate their tasks, an insight into this activity can provide them with a mean from which to develop and improve a teaching-learning context that stimulates learners and enhances the construction of knowledge (Johnson, 2007).

1.2 Modelling learners' activity in online discussion forums

In a wide sense, learner participation has been pointed out as a complex and intrinsic part of learning (Garrison, 1989; Wenger, 1998). From an online educational perspective, online learner participation can be defined as a process of learning that consists of taking part and maintaining relations with others, that comprises doing, communicating, thinking, feeling and belonging, and that occurs both online (*e.g.* through computer-mediated communication with peers and teachers) and offline (*e.g.* by studying course materials) (Hrastinski, 2008). As long as computer-mediated communication tools have been used in online teaching-learning environments, both understanding and encouraging online learner participation through the asynchronous interactions that rely in online threaded discussion forums have become major research issues (Bento and Schuster, 2003).

The next steps are dedicated to present, after having previously defined the specifics of their conceptual frame (section 1.2.1), what different approaches and strategies have been followed throughout the literature in order to model learners' activity in online discussion boards (sections 1.2.2 and 1.2.3).

1.2.1 Levels of participation and units of analysis

As a result of a literature review over 36 different works on the issue –from Ross (1996) to Caspi et al. (2008)–, Hrastinski (2008) defines a total of six levels of participation (pages 1756–1757) and categorises seven different units of analysis (pages 1758–1760) that are considered from different approaches adopted in literature on conceptualising what online learner participation in discussion forums is and how it can be modelled.

1.2.1.1 Conceptualisation levels of online learner participation in discussion forums

The different conceptions of what online learner participation in discussion forums is can be organised in a taxonomy composed of the six following levels:

1. **Participation as accessing e-learning environments**

   From its most elemental conception, participation can be understood just as the number of times the learner accesses the e-learning environment or the communication tool; *i.e.* the larger the access rate, the higher the degree of participation (Davies and Graff, 2005).

2. **Participation as writing**

   A second and more sophisticated conception determines that participation is equalled with writing; *i.e.* the degree of participation is directly proportional to the amount of writing ac-

tivity (number of words, notes or messages written by learner) (Mazzolini and Maddison, 2003).

3. **Participation as quality writing**

   One step further, participation can be conceptualised as the quality of the writing contributions; *i.e.* writing many contributions of high quality points to a more active and better participation. In this sense, a qualitative analysis of the content of the contributions can be performed and different types of statements (Davidson-Shivers et al., 2001) and/or messages (Kim et al., 2011) can be defined.

4. **Participation as writing and reading**

   Considering the two different kinds of activity that can be essentially performed in online discussion forums, the next level of conception states that participation is equalled with both writing and reading; *i.e.* learners that just read messages participate in a different way than learners that both read and write messages (Lipponen et al., 2003).

5. **Participation as actual and perceived writing**

   Once the reading activity performed by learners is taken into account, a more qualitative level of conception can be reached and both actual and perceived writing can be considered to determine what participation is; *i.e.* the learners' perception of the usefulness and importance of the messages written by a learner determines his or her level of participation (Vonderwell and Zachariah, 2005).

6. **Participation as taking part and joining in a dialogue**

   Finally, according to its last and most sophisticated level of conception, true participation occurs to the extent that learner is taking part and joining in a rewarding dialogue for engaged and active learning; *i.e.* social conceptions are adopted in the definition of participation, so that concepts like collaboration among participants and joint construction of learning are taken into account (Beuchot and Bullen, 2005).

1.2.1.2 Units of analysis on modelling online learner participation in discussion forums

Seven different units of analysis can be defined in order to determine how online learner participation is empirically studied:

1. **System accesses or logins**

   The simplest unit of analysis of learner participation is constituted by the system accesses or logins. This quantitative measure of participation simply takes into account how often learners access the forums where online discussions occurred (Davies and Graff, 2005; Caspi et al., 2008) or, even more generically, how many times learners log on the online learning environment (Ellis, 2003; Kuboni and Martin, 2004).

2. **Quantity of messages or items of information**

   The next unit of analysis is provided by another quantitative measure of participation, which is the amount of items of information produced by learner. Depending on the particular scenario and/or the available data, these items can be of different kinds; the most usual of them is the quantity of written messages (Ngwenya et al., 2008), but there are others, such as quantity of threads –sequences of reciprocal messages– (de Laat et al., 2007), ideas or reasonings (Hakkarainen and Palonen, 2003), complete statements or thoughts (Davidson-Shivers et al., 2001), and phrases or sentences (Böhlke, 2003).

3. **Message or item length**

   Being the empirical study of learner participation based on some kind of item of information (most typically, messages), the length of these items can also be used as a quantitative unit of analysis. This length can be defined as thread depth –number of messages or hierarchical levels in the thread– (Calvani et al., 2010), word counts (Woods and Keeler, 2001) or lines of information (Masters and Oberprieler, 2004).

4. **Message or item quality**

   In the same way than the item length, but from a qualitative perspective, the quality of the items of information can provide another unit of analysis. This item quality is obtained by categorising the items according to some kind of classification scheme. Different schemes and sets of categories can be defined from different approaches, such as on-topic and off-topic messages (Lipponen et al., 2002), new and reply messages (Beuchot and Bullen, 2005), statement, limited response, questioning response and dialogue posts (Sackville and Sherratt, 2006), question, answer, elaboration and correction messages (Ravi and Kim, 2007), different levels of critical thinking in messages (Bullen, 1998), or presence of key words and/or key phrases in messages (Garrison et al., 2000).

5. **Read messages**

   Whereas writing activity constitutes the explicit (or visible) way of participation in online discussion forums, reading activity is an implicit (or invisible) task and it can also be considered in order to define another unit of analysis of learner participation. Being the amount of read messages (Calvani et al., 2010) or how many times a learner read certain messages (Erlin et al., 2009) the most common quantitative measurements used within this unit of analysis, qualitative studies based on questionnaires (Takahashi et al., 2007) or surveys (Williams and Pury, 2002) can be used as well in order to characterise reading activity performed by learner.

6. **Learner perceptions**

   Evaluations or appraisals of any kind made by learners on how they perceive and consider participation are understood as learner perceptions and they can constitute another unit of analysis. Perceived participation can be measured and studied from different approaches,

including both quantitative (*e.g.* how many times and by how many learners a message has been read (Erlin et al., 2009)) and qualitative measures (*e.g.* interviews to learners (Bullen, 1998), reflective learner reports (Ellis, 2003; Vonderwell and Zachariah, 2005) or close-ended (Hrastinski, 2006) and open-ended (Kuboni and Martin, 2004) questions in surveys).

7. **Time spent**

   Finally, the temporal dimension of the activity performed by learners in online discussion forums can be taken into account in order to define a last unit of analysis of learner participation. The time spent by learners on participating in threaded discussions can be studied from both quantitative (*e.g.* measuring the rhythm of posting (Thomas, 2002; Calvani et al., 2010) or participation rates (Nandi et al., 2009)) and qualitative perspectives (*e.g.* using surveys to determine how many hours learners are engaged in online discussions (Hrastinski, 2006; McLinden et al., 2006) or the frequency and average length of their visits to the e-learning environment (Kuboni and Martin, 2004)).

More details on specific indicators used in literature in order to measure and characterise online learner participation can be found in (Dringus and Ellis, 2005, pages 149–150).

1.2.2 Modelling learners' activity from a theoretical perspective

Theoretical approaches to this issue try to provide conceptual frameworks, descriptive taxonomies and argumentative structures in order to explain and model learners' activity (Kelly, 2004). Nonetheless, both quantitative and qualitative analysis of the results of some field experiments are also performed in some works with the aim of giving empirical support to the theoretical proposals. According to the learning theories considered to study and describe the activity carried out by learners in online discussion forums, different kinds of theoretical approaches to this matter are observed in literature.

1.2.2.1 The behaviourist approach

From this kind of approach, learner's behaviour is modelled by considering the most basic and elemental actions that learners can come to perform in online discussion forums, which are essentially two: writing and reading. Dissimilarities among the distinct proposals made by several researchers according to this premise are, at the most, subtle, since they always end up defining the same three different models of behaviour.

In the first place, Mason (1994) finds that learners fall into three distinct groups in their online participation: **active participants** (those who both read and post messages), **lurkers** (those who read, but do not post messages), and **those who do not take part** (neither read, nor write messages).

In an analogous manner, three main types of learner can be defined regarding their participation in online discussions (Hammond, 1999): **the communicative learner**, who finds both the time to take part and the confidence to send messages; **the quiet learner**, who finds time to take part by reading messages but not to contribute and **the non-participant** who find time and other constraints impossible to overcome, even to the extent of reading messages. Hammond (1999) also points out that these three types are abstractions and, being learners able to slip from one to another –in particular, from very quiet periods of participation to more communicative periods–, this conception of the styles of participation can be helpful to show the patterns of behaviour that many learners adopt at least for significant periods of time.

But it is Taylor (2002) who, as a result of his investigation on learners' participation patterns in accessing and contributing to online discussions and their influence over academic achievement, propose one of the most popular terminologies for these three models of behaviour:

- **Workers**. The workers (or proactive participators) are learners who visit online forums regularly and contribute an above average number of postings to the threaded discussions. These proactive learners are continuously involved in discussions and are often among the first both to post a message and to respond quickly to other messages, thereby creating threads of ongoing dialogue among other peers.
- **Lurkers**. The lurkers (or peripheral participators) are learners who, even visiting online forums regularly, contribute occasionally to the threaded discussions (with less than the average number of postings), since their regular participation is mostly carried through in "read-only mode".
- **Shirkers**. The shirkers (or parsimonious participators) are learners who occasionally visit online forums and contribute a testimonial number of postings to the threaded discussions.

Slight nuances among some definitions proposed by different authors can be observed. Whereas Mason (1994) and Hammond (1999) define lurkers and quiet learners as only-readers, Taylor (2002) considers that lurkers may write, in addition, some occasional and peripheral contributions. In the same way, Taylor (2002) considers that shirkers can perform some marginal levels of participation, but Mason (1994) and Hammond (1999), maintain that, respectively, those who do not take part and non-participants are, as their names imply, absolutely passive learners who don't participate at all. The slight differences found in these definitions fit with the fact suggested by Taylor (2002), such that parameters for levels of learner participation should be defined –it is a difficult task, insomuch they are scenario-dependent–, so that the reasons for varying degrees of engagement can be unpacked.

In any case, it is the definition of lurking behaviour the one that transcend from the rest of models posed from this approach, since it have become one of the most interesting and deeply studied matters throughout the literature. A deeper insight into the distinctive features of lurking

behaviour can be reached by referring to contributions that tackle different subjects concerning lurking, such as giving a proper definition of lurking (Mason, 1994; Whittaker et al., 1998; Fritsch, 1999; Nonnecke and Preece, 2000; Salmon, 2000; Bowes, 2002; Taylor, 2002; Rafaeli et al., 2004), finding the specific characteristics of lurkers (Kollock and Smith, 1996; Morris and Ogan, 1996; Nonnecke and Preece, 1999; Preece et al., 2003), discovering the reasons that motivate lurkers to lurk (Salmon, 2000; Nonnecke and Preece, 2001; Beaudoin, 2002; Preece et al., 2003; Rafaeli et al., 2004), discussing how to move lurkers to posters (Whittaker et al., 1998; Bowes, 2002; Nonnecke et al., 2004; Bishop, 2011) and, probably the most complex of all, determining whether lurkers actually learn or not (Fritsch, 1999; Lee and McKendree, 1999; Salmon, 2000; Beaudoin, 2002; Rafaeli et al., 2004).

1.2.2.2 The social approach

The fundamentals of this approach lies on the definition of three capital concepts: impersonality, interpersonality and social presence. Impersonality refers to task-oriented communication in which information is offered or requested (Walther, 1996). Interpersonality includes social or personally oriented interaction, or informal communication, that leads to the creation of relationships among participants (Sudweeks and Simoff, 1999). The social presence –the ability of participants in a community of to project themselves both socially and emotionally, as real people, through the medium of communication being used– is closely related to interpersonality (Garrison et al., 2000). Thus, since research on asynchronous text-based online communication shows that it does permit high levels of interpersonal communication, learners' activity in online discussion forums can therefore be understood and modelled from the concept of social presence (Rourke et al., 1999).

From these premises, one of the most complete conceptual frameworks on modelling activity in asynchronous online discussions according to its social perspective is provided by Beuchot and Bullen (2005). Basing their proposals on the previous works of Lundgren (1977), Bales and Cohen (1979), Higgins (1991), Henri (1992), Walther and Burgoon (1992), Schutz (1994) and Mabry (1997), they define a double taxonomy in order to model, on the one hand, the nature of interaction among participants and, on the other hand, both the impersonal and interpersonal content of threaded discussions.

The 5-category taxonomy they propose to describe interaction is shown in Table 1.1 and it includes the detail of the corresponding descriptions for the five types of interaction they define. Next, the 13-category taxonomy utilised to categorise impersonality (two categories) and interpersonality (eleven categories) is shown in Table 1.2. The categories in this last taxonomy are associated each other by defining opposite –positive/negative– peers and are subsequently described in Tables 1.3 (impersonality categories) and 1.4 (interpersonality categories).

Types of interaction	Description
Active	Independence from and no reference to any previous sentence, message, person or group (*e.g.* introduction of a new topic).
Explicit reactive	Explicit reference to a previous sentence, message, person or group (*e.g.* direct answer to a previous question).
Implicit reactive	Implicit reference to a previous sentence, message, person or group (*e.g.* complement to an existing answer to a previous question).
Engaging interactive	Obvious attempts at engaging others at conversation (*e.g.* asking questions, suggestions, asking for comments, etc.).
True interactive	Any reference to the nature of previous sentences or messages (*e.g.* remarking on how a previous message is supportive, argumentative, etc.).

Table 1.1: Interaction: a 5-category taxonomy (Beuchot and Bullen, 2005).

	Categories	
	Positive	**Negative**
Impersonality	Informing-Offering	Asking-Requesting
Interpersonality	Support-Alignment	Adversariality-Opposition
	Disclosure	Reserve
	Appraisal	Chastisement
	Humour	Sarcasm
	Inquiry	Self absorption-Advocacy
	Others	

Table 1.2: Impersonality and interpersonality: a 13-category taxonomy (Beuchot and Bullen, 2005).

Category	Description
Informing-Offering	Forwarding factual information, either as a spontaneous offer, an answer, a question or a request.
Asking-Requesting	Asking for factual information; requesting repetition, clarification or confirmation; asking for examples; requesting data.

Table 1.3: Description of categories for impersonal content (Beuchot and Bullen, 2005).

From this conceptual framework on, social roles adopted by learners throughout their activity in online discussion forums can be defined and modelled. Along these lines, Bento et al. (2005) propose four elemental models of learners' roles, by simply considering low and high levels of both impersonality and interpersonality, as shown in Table 1.5.

Active learners and **social participants** come to be highly visible students, who often participate in online discussions and fundamentally differ in terms of their interaction with course contents (impersonality); **witness learners** actually interact with course contents but, since their low level of interpersonality, are invisible students and can be easily equalled to lurkers (see section 1.2.2.1);

Category	Description
Support-Alignment	Acceptance of points of view, agreement, approval, concession, compliance, friendly greetings, etc.
Adversariality-Opposition	Direct opposition, intellectual conflict, disagreement, critical view, judgement, assessment, etc.
Disclosure	Self-presentation; revelation of novel, ordinary or personal information apart from the subject of discussion.
Reserve	Appealing to end the discussion or no attempting to pursue it further; inhibiting the interaction or the development of an idea.
Appraisal	Admiration, commendation, praise, positive reinforcement of others' contributions, satisfaction with others' ideas, etc.
Chastisement	Anger, open hostility, personal attacks, disliking, rudeness, provocation, unfriendly and destructive comments, etc.
Humour	Explicit joking statements, display of wit, positive irony, tension relieving comments, use of puns and humorous language, etc.
Sarcasm	Derision, making fun of somebody or someone's ideas, cruel forms of humour, hostile wit, etc.
Inquiry	Asking expansive questions, asking for others' opinions, inviting peers to engage further in the discussion, requesting elaboration.
Self absorption-Advocacy	Forwarding opinions, self-centred use of personal pronouns, self-promotion, strong and forcefully worded assertions, etc.
Others	Any sentence non-assignable to any of the previous categories.

Table 1.4: Description of categories for interpersonal content (Beuchot and Bullen, 2005).

	Low impersonality	**High impersonality**
High interpersonality	Social participants	Active learners
Low interpersonality	Missing-in-action learners	Witness learners

Table 1.5: Taxonomy of impersonal and interpersonal participation in online courses (Bento et al., 2005).

and, finally, **missing-in-action learners** are passive students that care neither about the course content nor their peers' learning (Bento et al., 2005).

Beyond these basic models and since they are highly context-dependent, a great diversity of social roles presented in online discussion groups (*e.g.* local experts, answer people, conversationalists, fans, discussion artists, flame warriors, trolls, etc.) have been identified and defined, primarily by conducting ethnographic studies of the content of interaction (Donath, 1996; Burkhalter and Smith, 2004; Herring, 2004; Marcoccia, 2004). A relation of several roles proposed by different authors is shown in Table 1.6:

Authors	Roles
Kim (2000)	Visitors, novices, regulars, leaders and elders.
Golder (2003)	Newbies, celebrities, elders, lurkers, flamers, trolls and ranters.
Brush et al. (2005)	Key contributors, low volume repliers, questioners, readers and disengaged observers.
Waters and Gasson (2005)	Initiators, contributors, facilitators, complicators, knowledge-elicitors, vicarious-acknowledgers, closers and passive-learners.

Table 1.6: Different learners' social roles defined by various authors (Turner and Fisher, 2006).

1.2.2.3 The constructivist approach

The main goal of this kind of approach is to provide a description of learning that may come from asynchronous discussion. Both social cognitivism and constructivism are theoretical frameworks that might support asynchronous discussion (Tam, 2000), since the online classroom has been defined as a learner-centred environment (Knowlton, 2000) that allows knowledge construction among learners (Jonassen et al., 1995). Considering that constructivism emphasises descriptions of how learning occurs, the use of a constructivist frame seems therefore to be appropriate to model and explain learners' activity in online discussion forums (Bannan-Ritland, 2002).

Thus, taking the previous proposals of many authors as a starting point and being consistent with them, Knowlton (2005) puts forth, from a constructivist view of asynchronous discussion, a five-tiered taxonomy that defines five different kinds of participants. In addition, it is also provided a description of both the nature of participation in online discussion forums and the possible types of learning that derive from it:

- **Passive participants**. Some learners take a passive approach toward asynchronous discussion (Weedman, 1999). Passive participants –also known as lurkers– read contributions to the discussion, but they do not participate. Although there is a deficiency of knowledge about the real behaviour of passive participants (Graham and Scarborough, 1999), the reasons for their passivity might be, among others, a lack of understanding of the environment, a belittling of the role of asynchronous discussion in meeting content-based course goals, an attempt to understand confusing guidelines from the course instructor or ambiguous contributions from classmates, or even a consequence of their thinking styles (visual thinkers may tend to participate less than, for instance, analytic thinkers) (Lave and Wenger, 1991). Passive participants either do not value, or do not understand how to engage in, collaborative processes. Since they may be accustomed to a teacher-centred view of education, they may be comfortable by mirroring the instructor's knowledge and therefore do not need and/or do not know how to collaborate (Hara and Kling, 2000). However, regardless the real reasons of their passive attitude, passive participants certainly see knowledge as something constructed by others (Speck, 1998). They consider their own role as one of absorbing, so that they

take a product view –instead of a process view– of knowledge construction. For the passive participant, the process is not worthy of being shared with classmates.

- **Developmental participants**. Developmental participation exhibits more active behaviours than passive participation, but the contributions of developmental participants still do not substantively add to a real collaborative knowledge construction (Tam, 2000). Participants operating at this level understand that meaningful interaction can occur in asynchronous discussions, but they may see it in terms of the gains that they get from participating in, instead of the quality of their true learning about course content. Although they are no completely silent, developmental participants usually do not contribute anything new to the content-based discussion, only tangential reactions to contributions of others, and they rarely collaborate to a socially-negotiated construction of knowledge (Jonassen et al., 1995). They may understand personal aspects of collaboration such as community development (Palloff and Pratt, 1999) or social reinforcement (Prestera and Moller, 2001), but they dismiss, or simple have yet to discover, the educational enrichment than can come from online discussions. Developmental participants do not view asynchronous discussions as a place for the creation of new knowledge, but as a place where knowledge can be shown and exemplified. They participate to validate whether they correctly understand the facts, rather than to construct knowledge; they look for encouragement and reinforcement, rather than true discussions and opportunities to build which is not yet known, so that their attempts to participate may show some type of intrapersonal interaction or internal dialogue (Berge, 1999).

- **Generative participants**. Generative participation is characterised as a participant's attempt to offer commentary about course content. Participants at the generative level view the asynchronous discussion as a conducive environment to articulate ideas and the process of knowledge construction as a private and solitary act. Generative participation can actually lead to knowledge construction. Generative participants recognise the interactive possibilities of online discussions, but they do not view interaction as an educational necessity. They understand the environment as a mind tool and as a place to ask to the instructor and to report what they know, but not as a tool for distributing and sharing their ideas (Nicholson and Bond, 2003). Because of their teacher-centred view of the learning process, generative participants do not consider themselves as a part of a dialogue with classmates; instead of that, they are engaged in a type of collaboration with the instructor, which they connect to the target of earning a grade. With all that, the generative participant may at least have developed some kind of sense of communal confidence, since they trust that classmates will treat their contributions with respect (Rovai, 2001). Knowledge construction by means of generative participation can be seen both as a sort of generative learning –contributions are offered to discussions of response to a prescriptive request from a teacher– (Wittrock and Alesandrini, 1990) and as way of generating content and therefore constructing knowledge through the act of writing –since writing is a learning activity– (Adams and Hamm, 1990).

- **Dialogical participants**. Dialogical participation involves a substantive interaction among learners through asynchronous discussion. Dialogical participants recognise themselves as a part of a community of learners in the extent that they accept the mutual relation of content and context, of individual and environment, and of knowing and doing (Barab et al., 2001). They also understand that the environment itself allows knowledge to be constructed and they view it as a medium for increasing comprehension and focusing on task completion (Walther, 1996; Jonassen and Kwon, 2001). Dialogical participants consider collaboration as a source for knowledge construction and they can use and structure the environment to facilitate a stronger collaboration and, thus, a more durable knowledge construction (Weiss, 2000). Participants at the dialogical level tend to perform a strong use of asynchronicity, since they reflect on previous contributions to a discussion before they respond to those contributions (Lewis et al., 1997) and they recognise the potential for integrating past contributions to a topic as key benefit of the environment (Prestera and Moller, 2001). Dialogical participants understand the value of interacting with classmates about their thoughts and reasonings, instead of simply consume their ideas. They are not solitary thinkers, since they use the processes of debate and discussion both to encounter new ideas about course materials and to test their ideas through discussion with other participants (de Haan, 2002). The strong feelings of community among learners created and reinforced through dialogical participation can contribute to increase the flow of information and, thus, the efficiency of the knowledge construction process (Rovai, 2001).

- **Metacognitive participants**. Metacognitive participation is strongly connected to internal mental representations of the learning process. In this way, the constructions of knowledge distributed among a group of learners that take place through asynchronous discussions are understood and interpreted by participants at the metacognitive level in direct relationship to themselves. Through online discussions, metacognitive participants are able to learn about themselves and their own learning processes (Knowlton, 2003b) by performing a sort of reflective learning (Berge, 2002). Thus, by viewing their own contributions to online discussions as good and strong, they are likely to learn better and more than those who do not have positive regard for the role that they play in asynchronous discussions (Leinonen and Järvelä, 2003). Furthermore, in the extent that they use the online forums to monitor their own comprehension of the discussion contents, metacognitive participants understand the asynchronous environment as a place conducive both to learn about the content of the discussion and to learn about the self (Lin, 2001). They use the archives of the discussions not solely to make reconsiderations on course contents, but they are also focused on the relationship between their current and previous perspectives and thoughts. In order to development of strategies for generating knowledge, the metacognitive participant may ask other participants about the strategies they follow to understand and interpret concepts or assignments, which can contribute to all learners' monitoring and awareness of their own knowledge (Knowlton, 2003a). And, finally, metacognitive participants understand that

knowledge construction transcends the boundaries of asynchronous discussion, since their are able to reflect on how knowledge construction occurred for them both throughout the discussion and beyond the discussion itself (Palloff and Pratt, 1999).

Finally, it is worth noting that, in spite of the fact that the conceptual framework described from this taxonomy is generic, the nature of asynchronous discussions –and therefore the way learners participate in– surely depends on the specific field of study; *i.e.* the types of both participation and learning produced from different domains –such as hard sciences, social sciences or humanities– can easily be dissimilar from each other (Dunlosky, 1998; Knowlton, 2003a,b).

1.2.3 Modelling learners' activity from the data mining paradigm

Data mining (DM) is defined as the application of specific algorithms for extracting patterns (or models) from data –*i.e.* making any high-level description of a set of data, such as, fitting a model to data or finding structures from it– (Fayyad et al., 1996). As shown in Figure 1.1, DM can be seen as an integral part of knowledge discovery in databases (KDD), which can be generically defined as the overall process of converting raw data into useful information (Tan et al., 2006).

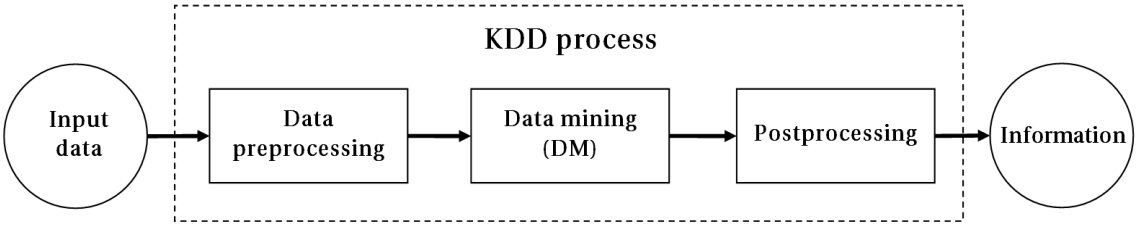


Figure 1.1: DM as an integral part of the KDD process (adapted from Tan et al. (2006)).

Since most DM-based studies are performed in a particular application domain, common previous steps in this process consist in stating the problem to solve and collecting the data (Kantardzic, 2003). In general terms, the input data (or target data set) is constituted by a set of objects (or elements, or points) represented upon a space of variables (or attributes, or features, or dimensions), which can be either numerical –representing the value of a quantitative measurable magnitude– or nominal –taking one of a predefined set of categorical values– (Sevillano, 2009).

Thus, on the one hand, transforming the raw input data into an appropriate format for subsequent analysis is the purpose of the data preprocessing stage, which can entail transformation processes such as noise removal, attributes scaling, data parametrisation, or dimensionality reduction (by means of feature selection and/or feature extraction techniques), among others. On the other hand, the data postprocessing stage basically consists in evaluating and interpreting the patterns and models obtained from the DM stage and it often involves processes such as calculation of statistical measures, application of hypothesis tests, or visualisation of the mined patterns, among others (Tan et al., 2006).

The core of the KDD process is the DM stage, where the patterns and models that lead to useful information are obtained from the preprocessed data. Depending on the goals of the KDD process, a suitable DM task must be identified and, consequently, a proper DM method must be chosen. According to Fayyad et al. (1996), DM tasks can be classified into **verification-oriented tasks**, whose goal is to evaluate a previously proposed hypothesis, and **discovery-oriented tasks**, whose goal is to discover useful patterns in the data. For their part, both prediction and description tasks can be distinguished among the discovery-oriented tasks, as shown in Figure 1.2. The goal of DM **predictive methods** –classification and regression– is to find patterns in order to predict the future behaviour of some entities Fayyad et al. (1996), whereas the different families of DM **descriptive methods** –clustering and summarisation, among others– are often more exploratory in nature, since they focus on finding understandable representations of the underlying structure of the data (Maimon and Rokach, 2005).

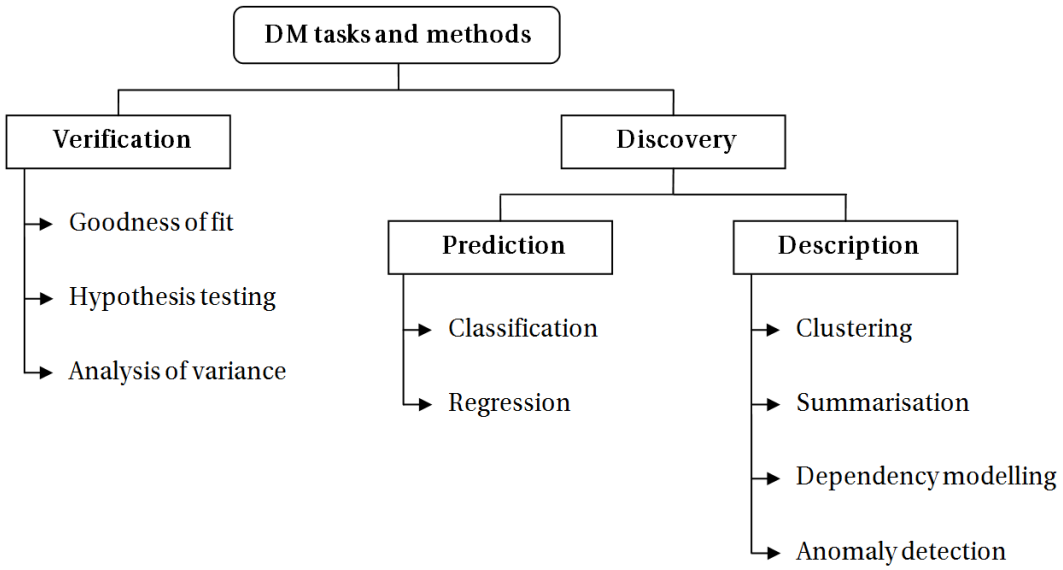


Figure 1.2: A taxonomy of DM tasks and methods (adapted from Sevillano (2009)).

Once a proper DM method is chosen and taking into account which parameters are the most appropriate from an algorithmic viewpoint, as well as which levels of accuracy, utility and intelligibility are required for the patterns and models of the data, a specific algorithm must be selected to be applied at the DM stage of the KDD process (Fayyad, 1996). Regarding this issue, it is worth noticing that extracting knowledge from a data set is a both multistage and iterative process, since the interpretation/evaluation of the obtained patterns performed in the postprocessing stage can lead to re-execute any of the previous stages for further refinement of the KDD process (Brachman and Anand, 1996; Halkidi et al., 2002a).

From both conceptual and practical viewpoints, KDD and DM are interdisciplinary fields that bring together researchers, developers and practitioners from a wide variety of related fields, including statistics, machine learning, artificial intelligence, databases management, knowledge acquisition, pattern recognition, information retrieval, visualisation, intelligent agents for distributed

and multimedia environments, digital libraries and management information systems (Fayyad, 1996). In this way, there are many areas where DM techniques are applied, such as finance and business, marketing, social sciences, medicine and healthcare, monitoring and diagnosis, science data acquisition, manufacturing, engineering, telecommunications industry and automotive sector, among many others (Fayyad et al., 1996; Kantardzic, 2003; Tan et al., 2006). Thus, the application of the principles of this paradigm from the educational scope gives rise to the educational data mining (EDM) field (Romero and Ventura, 2007).

EDM is a discipline concerned with developing methods for exploring the unique types of data that come from educational settings and using those methods to better understand learners and the settings which they learn in (Baker and Yacef, 2009). Making use of the diversity of DM methods and algorithms, the aim of EDM is to analyse educational data to find out descriptive patterns and predictions that characterise learners' behaviours and achievements, domain knowledge content, assessments and educational functionalities and applications (Romero and Ventura, 2007). EDM is applied to address a wide variety of goals, which can be encompassed by the following general applications: communication to teachers and instructors, maintenance and improvement of courses and environments, generation of recommendations, modelling of learning behaviours, prediction of learners' grades and learning outcomes, and analysis of domain structures (Romero et al., 2011).

Therefore, the matter of modelling learners' activity in online discussion forums is addressed from the EDM field in the terms of a modelling problem of learning behaviours. Unlike the theoretical approaches previously described, the DM-based approach tackles this issue from an empirical angle: the activity performed by learners in online discussion forums generates a flow of measurable and collectable data that is analysed in order to discover and identify the different learning behaviours and roles adopted by learners. The literature in this specific area indicates that this modelling problem is mainly posed either from a social network analysis perspective (section 1.2.3.1) or as a clustering scenario (section 1.2.3.2).

1.2.3.1 Modelling learners' activity by means of social network analysis techniques

The social network analysis (SNA) is a widely used research methodology in modern sociology and anthropology for the study of the social relationships between individuals in a community and it is also utilised to analyse interaction among members of a virtual community (Garton et al., 1997; Cho et al., 2002; Reffay and Chanier, 2003). Moreover, there is a line of work on applying SNA for modelling learners' activity in online forums, based on extracting social networks from asynchronous discussions and finding appropriate indicators for measuring and evaluating learners' participation (Lipponen et al., 2003; Willging, 2005; Laghos and Zaphiris, 2006; de Laat et al., 2007; Welser et al., 2007; Erlin et al., 2009; Calvani et al., 2010; Sundararajan, 2010; Rabbany et al., 2011).

Since SNA seeks to identify underlying patterns of social relations based on the way actors are connected with each other in a social network (Scott, 1991; Wasserman and Faust, 1994), it is a suitable choice in order to assist in describing and understanding the patterns of participation and interaction among learners in online discussion forums (de Laat et al., 2007). SNA is therefore a specially useful technique in order to find and identify social roles (see section 1.2.2.2) in asynchronous online discussions (Welser et al., 2007). Thus, interactions among participants in discussion boards can be properly mapped out and explored using SNA, since it provides analytical data about the activity and relationships of the learners (de Laat et al., 2007), as well as useful visualisations of participants' activities (Calvani et al., 2010).

In this way, SNA can be somehow regarded, from a DM perspective, as a summarisation method (see Figure 1.2 in section 1.2.3), since it makes use of multivariate visualisation methods with the aim of providing a compact description for the data (Rabbany et al., 2011). Furthermore, SNA is usually utilised in combination with other different techniques, such as content analysis and critical event recall (de Laat et al., 2007), or regression techniques (Welser et al., 2007).

Using SNA, the social environment developed throughout asynchronous discussions in online forums is mapped as patterns of relationships among interacting learners (Wasserman and Faust, 1994). The focus is placed on the relational data, instead of the characteristics of each single learner. Thus, the unit of analysis in SNA is not the individual, but the interaction that occurs between members of the network. SNA allows to visualise the network of relations among learners based on the interpersonal activity (see section 1.2.2.2), *i.e.* by means of the presence and absence of connections between them (de Laat et al., 2007).

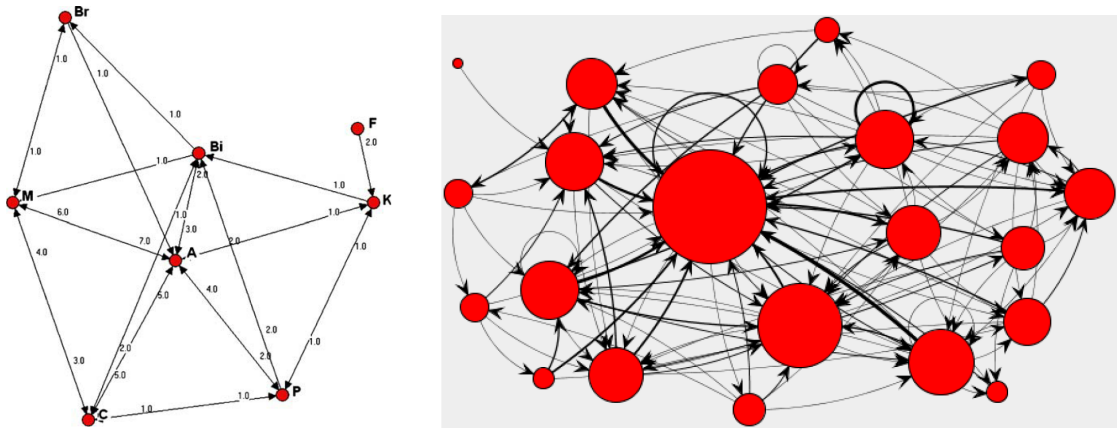
The two key indicators of SNA are density and centrality (Lipponen et al., 2003). On the one hand, density provides a measure of the overall connections among participants. The density of a network can be defined as the ratio between the number of communicative links observed in a network and the maximum number of possible links. Therefore, the more participants connected to each other (*e.g.* by their exchange of messages), the higher will be the density value of the network (Scott, 1991). On the other hand, centrality is a measure that provides information about the behavior of individual participants within a network. Centrality indicates the extent to which an individual interacts with other members in the network (Wasserman and Faust, 1994). In order to, for instance, identify central and peripheral participants of a social network through this measure, the number of each participant's connections with other members is measured and the in-degree –the number of learners who respond to a message from a certain participant– and out-degree –the number of messages a learner sends to other peers in the online forums– centrality values of each participant are generated (de Laat et al., 2007).

Driven by these fundamentals, different ways of visualising learners' activity in online discussion forums are proposed from the SNA-based approach, the following being the most popular ones:

- **Sociograms**. The structure of interpersonal relations in a group situation is plotted in order to easily visualise both the distribution of the network's density and the centrality degree of each participant (Turner et al., 2005; de Laat et al., 2007; Welser et al., 2007; Erlin et al., 2009). Sociograms can be useful to study both classroom dynamics (interaction patterns within a group of learners) and the in-degree and out-degree activity around a single participant (ego networks). Different examples of sociograms are shown in Figure 1.3.

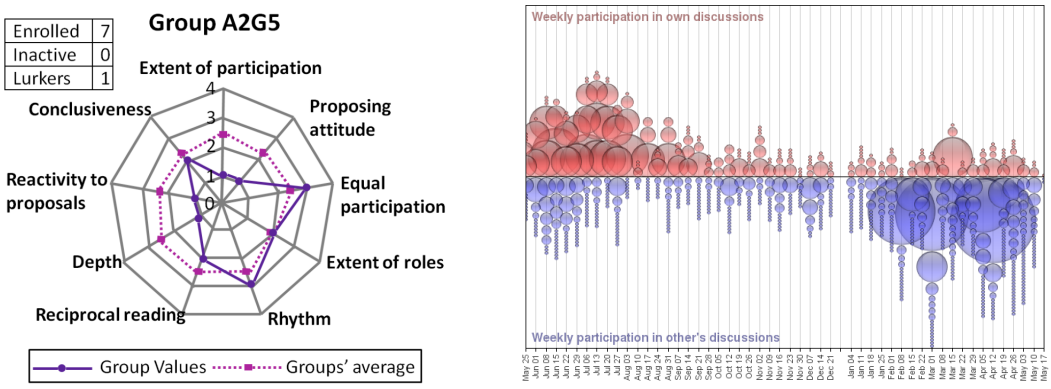
- **Radiant graphs**. Given a set of measurable and quantifiable variables (or indicators), the behaviour of both a group of learners or a single participant is visualised in comparison with the average behaviour of the classroom (Calvani et al., 2010; Rabbany et al., 2011). Radiant graphs can be useful both to illustrate specific behaviours and to evaluate learners participation in the context of a classroom. An example of radiant graph is shown in Figure 1.4a.

- **Authorlines**. The volume of contributions for a single learner is visualised in a temporal series form (Viégas and Smith, 2004; Turner et al., 2005; Welser et al., 2007). Authorlines can reveal detailed patterns about learners' posting/reading behaviour through time. An example of authorline is shown in Figure. 1.4b.



(a) An example of link-weighted sociogram. *(b)* An example of node- and link-weighted sociogram.

Figure 1.3: SNA visualisations: sociograms (extracted from de Laat et al. (2007) and Rabbany et al. (2011)).



(a) An example of radiant graph. *(b)* An example of authorline.

Figure 1.4: SNA visualisations: radiant graph (extracted from Rabbany et al. (2011) and Turner et al. (2005)).

Thus, several recent works in the literature show how to visually identify and characterise, by means of SNA techniques, a great diversity of participation roles in a variety of asynchronous discussion contexts (as previously mentioned in sections 1.2.2.2 and 1.2.2.3, participation roles are highly context-dependent).

Turner et al. (2005) analyse patterns of activity and development in a worldwide distributed Internet asynchronous discussion system called Usenet[1]. By a series of visualisations (treemaps[2], authorlines and sociograms, among others) that allow to categorise newsgroups, authors and threads, they identify eight different social roles performed by Usenet participants: **one-time-only posters**, **questioners**, **answer persons**, **conversationalists**, **trolls**, **spammers**, **binary posters** and **flame warriors**.

By combining the use of content analysis, critical event recall and SNA visualisations (sociograms), de Laat et al. (2007) analyse interaction patterns in small groups of learners and discuss the relation between **central/peripheral participation** and active/passive learning. Their conclusions suggest that the most active (or central) participants do not necessarily regulate and dominate the discussions, since while some learners are more socially engaged, others show more extensive metacognitive skills (Hara et al., 2000). Furthermore, having observed variations through time in the centrality of participants, they also find that learners may develop different roles or interests during their collaborative work (Reuven et al., 2003).

Analysing data drawn from Usenet[1] posts, Welser et al. (2007) obtain both sociograms and authorlines that allow to identify the patterns of contribution (or signatures) of three main user roles: **question person**, **discussion person** and **answer person**. By combining the SNA approach with some regression techniques, they conclude that the obtained signatures are strongly correlated with role behaviours and may lead to a predictive model for identifying participation roles.

In the specific context of online political discussions, Himelboim et al. (2009) analyse patterns of thread initiation and reply in several political newsgroups collected from the Microsoft Research Netscan dataset[3]. By combining the use of sociograms and content analysis, they are able to identify two different kinds of participants that act as discussion catalysts: **content importers** and **conversation starters**.

Finally, this brief review ends with the work of Calvani et al. (2010), who study the effectiveness of the interactions among learners in small online collaborative groups. They propose to characterise the interactions performed in each group of learners by means of nine different variables: extent of participation, proposing attitude, equal participation, extent of roles, rhythm, reciprocal readings, depth, reactivity to proposals and conclusiveness. This set of variables constitutes the representation space where, by means of radiant graphs, the interaction pattern of each group is

[1] http://www.usenet.org
[2] An example of treemap is available online at http://jcmc.indiana.edu/vol10/issue4/turner.1b.gif
[3] http://research.microsoft.com/en-us/groups/scg

visualised and put in comparison with the average interaction pattern of all groups. An example of radiant graph for a scarcely effective group can be observed in Figure 1.4a.

1.2.3.2 Modelling learners' activity by means of clustering techniques

Clustering (or cluster analysis) can be defined as the process of separating a finite unlabelled dataset into a finite and discrete set of natural clusters based on resemblance (Jain et al., 1999; Xu and Wunsch II, 2005). Being one of the most common DM descriptive methods, the goal of cluster analysis is to identify the underlying structure of the data and to represent it by means of a cluster-based descriptive taxonomy (where similar objects are labelled with the same cluster labels) that characterises the data with no previous knowledge (Sevillano, 2009).

Cluster analysis seeks to find clusters (or groups) of closely related objects in the data, so that those objects belonging to the same cluster are more similar to each other than those belonging to other clusters (Tan et al., 2006). In this way, the concept of cluster can be understood as a set of entities which are alike, whereas entities from different clusters are not alike (Everitt et al., 2011).

Clustering is, thus, an unsupervised task, since the objects in the dataset are unlabelled –*i.e.* there is no prior knowledge about how they should be grouped– and the process of identification of the clusters is data-driven –*i.e.* the cluster labels are obtained solely from the data, not provided by an external source– (Jain et al., 1999). This absence of category labels that tag objects with prior identifiers is what distinguishes cluster analysis from supervised tasks such as classification (see Figure 1.2). Unlike to supervised techniques, cluster analysis is geared toward finding existent structures in the data in order to find a convenient and valid organization of the data, not to establish rules for separating future data into categories (Jain and Dubes, 1988).

After a clustering process, the objects contained in the dataset can be represented by a set of meaningful clusters that define a simplified cluster-based data model, since the number of clusters is comparatively smaller than the number of objects (Berkhin, 2006). However, the obtained clusters should somehow reflect the mechanisms that cause some objects to be more similar to each other than to the remaining ones (Witten and Frank, 2005).

Being such a generic DM task, clustering-based applications can be found in a wide variety of research fields, such as psychology and other social sciences, economics, climatology, biology, computational genomics, statistics, education, information retrieval, text mining, computer vision and machine learning, among others (Tan et al., 2006). Focusing on the EDM field, three main categories of studies that deal with clustering problems in e-learning contexts can be found in literature: works that group e-learning material based on their similarities, works that group learners according to their navigational and/or learning behaviour, and works that use cluster analysis as part of an e-learning strategy but do not present any practical application results (Vellido et al., 2011). Thus, being directly related with the analysis of learning behaviours, modelling learners'

activity in online discussion forums by means of cluster analysis clearly fits in the second group of works.

From a generic perspective and based on the KDD fundamentals (see section 1.2.3), the process of modelling learners according to their learning behaviours by means of cluster analysis is composed of three main stages (see Figure 1.5):

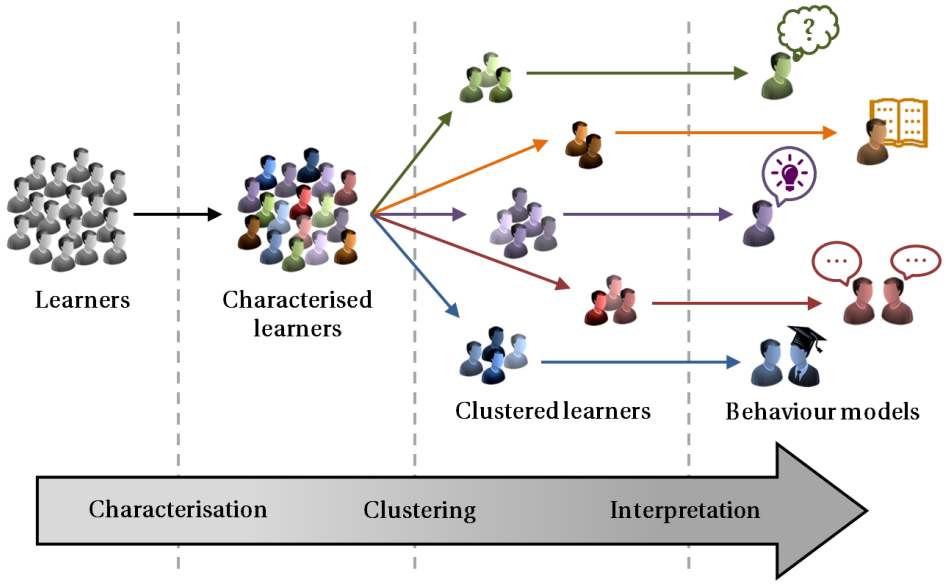


Figure 1.5: Modelling of learners' behaviour by means of cluster analysis.

- **Characterisation**. Raw data from learners' activity is first collected and then preprocessed. The result of this preprocessing process is a set of features that represent and characterise learners in terms of their behavior regarding the activity subjected to analysis (Amershi and Conati, 2009). The features that can be used for characterising learners' behaviour always depend on both the type of activity and the available data, but, in the specific case of modelling learners' activity in online discussion forums, different levels of participation and units of analysis have been defined to that effect (see section 1.2.1).

- **Clustering**. The characterised learners are then used as input objects to a clustering algorithm that groups them according to their resemblance (or proximity), *i.e.* learners with similar behaviour patterns in terms of the features chosen in the previous stage are grouped together in the same cluster (Amershi and Conati, 2009). This stage involves taking decisions that may greatly influence the outcomes of the modelling process (Sevillano, 2009), such as choosing the specific clustering algorithm to be applied, selecting the proximity measure, or determining the optimal number of clusters in the data (see Chapter 2 for further details).

- **Interpretation**. Each resulting cluster represents learners who, regarding the activity subjected to analysis, behave both similarly among them and dissimilarly among the rest of learners belonging to other clusters (Amershi and Conati, 2009). The obtained set of clusters is then analysed in order both to validate the quality of the clusters (see section 2.4 for further details), and to interpret what learning behaviour is represented by each cluster of

learners and its meaning according to the activity subjected to analysis (although both matters –validation and interpretation– may become strongly related). This interpretation of the resulting behaviour models has to be performed according to features chosen in the first stage of the process and, in the specific case of modelling learners' activity in online discussion forums, considering the conceptual frameworks posed from the different theoretical approaches to the issue (see section 1.2.2).

Thus, following these fundamentals, clustering-based approaches are useful in building an understanding of learners' behaviours in many digital environments, *e.g.* profiling hypermedia users according to their navigational behaviour (Barab et al., 1997), characterising behaviour groups in unstructured collaborative spaces (Talavera and Gaudioso, 2004), analysing cognitive tool use patterns in a hypermedia learning environments (Liu and Bera, 2005), personalising learning paths in intelligent e-learning systems (Zakrzewska, 2008), building user models for exploratory learning environments (Amershi and Conati, 2009), or predicting learners' final marks in a university course (López et al., 2012), among many others. More details on the usage of cluster analysis in educational environments can be found in the complete survey performed by Vellido et al. (2011). Regarding the particularities of adopting clustering-based strategies in order to identity different learning behaviours according to the activity performed by learners in online discussion forums, the literature provides several recent works that tackle this issue.

Firstly, different approaches to learning in a learner-centred online environment are analysed by del Valle and Duffy (2009):

- **Input data**. Clickstream data along a complete course are collected from 59 learners belonging to different fields, such as arts, math, science and social studies, among others.
- **Characterisation**. Every learner is characterised by means of three variables regarding the activity carried out in online asynchronous discussions and other seven general variables:
  - Proportion of time spent in the message board
  - Number of messages sent to the teacher
  - Number of times each teacher's message is read
  - Total time spent in the learning environment
  - Course duration
  - Total number of sessions performed to complete the course
  - Average inter-session interval
  - Proportion of time on learning resources
  - Proportion of learning resources accessed
  - Excess of learning sessions beyond the regular course planning

- **Cluster analysis**. Following the proposals of Barab et al. (1997), the cluster analysis is performed by using **Ward's agglomerative hierarchical clustering algorithm** and comparing characterised learners by means of the square Euclidean distance. Being *a priori* unknown, the optimal number of clusters is manually determined by obtaining several clustering solutions (from one to sixty clusters) and comparing them according to the distance between pairs of clusters: an inflection point is observed between the three and four clusters solutions, so that the three clusters solution is finally chosen. Having obtained the final set of clusters after a manual and partially subjective criterion, meaningful differences in the characterisation variables along the three clusters are confirmed by performing several statistical analysis (a Kruskal-Wallis one-way ANOVA with cluster membership as the independent variable and a Mann-Whitney U pairwise comparisons to determine significance between pairs of clusters).

- **Results**. Three distinctive approaches to online learning are identified:

  - **Mastery oriented (or Self-driven)** approach, formed by learners (59.3% of the sample) committed to the course and self-driven in their work, with the highest number of work sessions, learning resources used and messages read.

  - **Task focused (or "Get it done")** approach, formed by learners (22% of the sample) mainly devoted to complete the course tasks as soon as possible, with an intermediate number of work sessions, total hours online and messages read, but the lowest number of calendar days invested in the course.

  - **Minimalist in effort (or "Procrastinator")** approach, formed by learners (18.7% of the sample) weakly committed to the course, with not very frequent logins, no regularity of work and a tendency to spread the work over time.

Next, the study performed by Bliuc et al. (2010) focuses more specifically on the conceptions of learning that learners have through both face-to-face and online asynchronous discussions:

- **Input data**. Data from close-ended questionnaires answered by 113 learners enrolled in a political science course of one semester long are analysed.

- **Characterisation**. Following Biggs (1987) and Crawford et al. (998a), three different types of questionnaires focused on three different matters are used:

  - Conceptions of learning through discussions (16 items)

  - Approaches to learning through face-to-face discussions (21 items)

  - Approaches to learning through online discussions (21 items)

  Learners are characterised by means of the 27 more relevant items according to a principal component analysis (PCA) performed in the preprocessing stage.

- **Cluster analysis**. The existence of distinctive groups of learners with different conceptions, approaches and academic performance is explored by means of a **agglomerative hierarchical cluster analysis**, using standardised Z-scores for the variables and the Euclidean distance as proximity measure. After conducting a manual inspection of the results (similar to the one performed by del Valle and Duffy (2009)) and a discriminant analysis between groups, two separate clusters of learners are obtained.
- **Results**. Two different types of learners are identified according to their conceptions of and approaches to learning:
  - Those who had a cohesive conception of learning and took a **deep approach to online discussions** (71.7% of the sample).
  - Those who had fragmented conceptions and took a **surface approach to online discussions** (28.3% of the sample).

On classifying discussion boards of general topics –not strictly educational– according to the predominant interaction profiles of their participators, Chan et al. (2010) identify several different user roles:

- **Input data**. Interactions among 2530 users, during a period of six months, along twenty different online forums of a variety of topics [4] are analysed.
- **Characterisation**. Nine different features are utilised to characterise every user's activity:
  - In-degree coefficient of user's ego-centric network (see section 1.2.3.1)
  - Out-degree coefficient of user's ego-centric network (see section 1.2.3.1)
  - Percentage of other users with reciprocal communication
  - Percentage of threads with reciprocal communication
  - Mean of written posts per thread
  - Standard deviation of written posts per thread
  - Percentage of other users with at least one reply to the user
  - Percentage of posts with at least one reply to the user
  - Percentage of threads initiated by user

  Users with marginal participation profiles are discarded after applying PCA.
- **Cluster analysis**. Users are grouped Euclidean distance is used as proximity measure) by means of an **agglomerative hierarchical clustering algorithm**. In order to determine the final number of clusters, several clustering solutions are obtained. After applying five different clustering validation techniques (Rand, Silhouette, RS, Root mean square and DB Index) and a manual inspection to the results, the 15 clusters solution is finally chosen.

[4] http://www.boards.ie

- **Results**. The meaning of every cluster is interpreted according to the values of its users' features and eight different user roles (atypically, some roles are represented with more than one cluster) are identified:
  - **Joining conversationalists**: no initiators; high communicators with a small set of users.
  - **Popular initiators**: high popularity and levels of thread initialisation.
  - **Taciturns**: low communications; limited conversations with a few users.
  - **Supporters**: relatively middle of the road statistics; typical average participants.
  - **Elitists**: low percentage of neighbours and high percentage of bidirectional threads, which indicate strong conversations with a very small set of users.
  - **Popular participants**: involved with a large percentage of users; low initiators.
  - **Grunts**: low communications to a few users; relatively high levels of reciprocity.
  - **Ignored**: very low percentage of posts get replied to.

Khan et al. (2012) present a recent study, in terms of participation in social learning, of the behaviour patterns associated with learners accessing an online discussion forum:

- **Input data**. 303 learners from undergraduate course on project management. The study covers two years (163 learners on the first year; 140 on the second) that are analysed separately.
- **Characterisation**. Three different features are utilised to characterise learners' activity:
  - Average interval between online work sessions
  - Average duration of an online work session
  - Average number of discussion messages read during an online work session
- **Cluster analysis**. Using Euclidean distance as a proximity measure, learners are clustered by means of **Ward's agglomerative hierarchical clustering algorithm**. After a manual inspection of the clustering results, two sets of 5 clusters are obtained (one set per year). Having applied subjective criteria to obtain the two sets of clusters, one-way ANOVA and Tukey post-hoc tests are performed in order to find significant statistical differences among clusters.
- **Results**. By combining the two resultant sets of 5 clusters (one set per year), seven different types of learners are identified:
  - **Strategic learners**: mainly asocial learners; short and infrequent work sessions.
  - **Apathetic learners**: mainly asocial learners; medium-length and infrequent work sessions.
  - **Detached learners**: mainly asocial learners; long but infrequent work sessions.
  - **Directed learners**: slightly social learners; short and medium-spread work sessions.

  - **Purposive learners**: slightly social learners; short and frequent work sessions.
  - **Inquisitive learners**: highly social learners; medium-length and frequent work sessions.
  - **Committed learners**: highly social learners; long and frequent work sessions.

And, finally, Wise et al. (2013) examine learners' participation patterns in online discussion forums with the goal of getting an insight into learners' online listening behaviours, *i.e.* how they engage with the posts contributed by other peers (Wise et al., 2011b, 2012):

- **Input data**. In the context of an undergraduate business course taught in a blended format, clickstream data is collected for 95 learners from 3 week-long online discussions to solve organisational behavior challenges in groups of 10-13 participants.
- **Characterisation**. Following their own proposals in previous works (Wise et al., 2011a), the different facets of every learner's listening and speaking activity in the discussion forum are characterised by seven different variables:
  - Average length of session
  - Percent of sessions with posting actions
  - Percent of posts viewed at least once
  - Percent of total views (reads) of others' posts
  - Average length of time reading a post
  - Average number of posts contributed per discussion
  - Average number of reviews of own posts per discussion

  In addition, six other variables are used for additional comparisons to further characterise the differences between the clusters:
  - Average number of sessions per discussion
  - Average number of reads before contributing a post
  - Average number of views per discussion
  - Average number of words per post
  - Average length of time creating a post
  - Final grade
- **Cluster analysis**. The squared Euclidean distance metric and **Ward's agglomerative hierarchical clustering algorithm** are utilised to determine the distances between clusters for possible solutions. Following the same process performed by del Valle and Duffy (2009), a three clusters solution is finally obtained. Meaningful differences in the variables along the three clusters are confirmed by performing several statistical analysis (one-way ANOVA and post-hoc analysis using Tukey's HSD criterion with a Bonferroni alpha level correction).

- **Results**. Three types of learners are identified according to their listening behaviours:
  - **Superficial listeners/Intermittent talkers** (31% of the sample), who have a moderate amount of brief sessions, show a modest breadth of listening, perform shallow readings, write a poor average of posts per discussion, and do not exhibit a great deal of reflectivity on their own postings.
  - **Concentrated listeners/Integrated talkers** (49% of the sample), who have a limited amount of extended sessions, show a modest breadth of listening, read posts in great depth, write a poor average of posts per discussion, and exhibit a limited reflectivity on their own postings.
  - **Broad listeners/Reflective talkers** (20% of the sample), who have a great amount of extended sessions, show comprehensive breadth of listening but only moderate depth on readings, write posts frequently, and exhibit a great reflectivity on their own postings.

Key issues: selecting the clustering algorithm and determining the optimal number of clusters

Having reviewed in detailed the works in literature that propose clustering-based strategies in order to model learners' activity in online discussion forums, two common aspects of their DM-based analysis processes reveal themselves as key issues: the selection of the clustering algorithm and the determination of the optimal number of clusters. While the first one is, considering the particularities of this modelling problem, both reasonably and properly settled, the second one is sub-optimally solved and it constitutes the main goal the present thesis is built up around.

On the one hand, given the nature of the problem, it is not happenstance that the totality of the reviewed works opt for choosing agglomerative hierarchical clustering methods. As it has been shown throughout the present chapter, identifying learners' behaviours according to their participation patterns in online asynchronous discussions is a context-dependent modelling problem of exploratory nature. The different learning behaviours that can be performed by learners depend on many variables, such as the total amount of students in the classroom, the duration of the course, the field of study (hard sciences, social sciences, maths, art, economics, etc.), the teaching-learning strategies promoted by teacher (group work, individual work, mandatory/recommended use of the discussion board, etc.), the kind of subject (theoretical/practical, fully online/blended, etc.), among others. The available input data and therefore the features that can be utilised to characterise learners' activity are, as well, another contextual factor that determine to a large extent what different learning behaviours may become identified and how deeply they may be described. Furthermore, being, from a DM-based perspective, a discovery-oriented problem of exploratory and descriptive nature, it requires of analysis methods that provide as much amount of information about the obtained models as possible. Hence, under these premises, agglomerative hierarchical clustering methods are clearly the most suitable option when posing the problem from a clustering scenario perspective (see Chapters 2 and 3 for further details).

On the other hand, a proper identification of the different learning behaviours performed by learners is closely related with and strongly dependent on a correct determination of the real number of clusters in the dataset (Xu and Wunsch II, 2005). Firstly, it has to be noticed that this issue constitutes by itself another factor to contemplate in the election of a suitable clustering algorithm (as shown in Chapter 3, hierarchical clustering methods provide useful information that, among other aspects, may assist in determining the number of clusters). Furthermore, all the reasons stated in the previous paragraph to justify the choice of the clustering method can be applied to this matter as well: it is impossible to know *a priori* how many clusters of learners there are in the dataset when dealing with such a variable and context-dependent scenario. Therefore, the clustering-based approach to the matter of modelling learners' activity in online discussion forums reveals the underlying problem of the determination of the number of clusters, which is one of the best-known issues of the clustering paradigm (see section 2.5.1 for further details).

As in many other clustering-based applications, the previously reviewed works in the present section perform a manual and partially subjective process in order to select the number of clusters and obtain a final clustering solution. Essentially, this process consists in generating a diversity of solutions, plotting the value of some clustering criterion (distance between pairs of clusters, sum of the square error with respect to the clusters centroids, etc.) against the number of clusters and manually seeking for some kind of knee, peak or inflection point that suggests a particular number of clusters (*e.g.* see del Valle and Duffy (2009), Chan et al. (2010) or Wise et al. (2013) for more details). This is clearly a sub-optimal strategy that may easily lead to sub-optimal results of the KDD process (Tan et al., 2006). This kind of approach has several important drawbacks such as, in a first instance, requiring of a diversity of clustering solutions to compare each other, which entails a more time-consuming process. In addition, it is not a completely data-driven process –which would be desirable and more appropriate–, since it necessitates the user to manually inspect the procedure and to take a final decision under subjective criteria, which may lead to a non-optimal clustering solution biased by user's prior expectations (Everitt et al., 2011). Moreover, in order to both complement and compensate the subjective dimension of the process, it also requires the use of validation methods and/or statistical tests, which may help in determining which one of the generated solutions fits better with certain statistical criteria (see section 2.4 for further details), but do not guarantee to identify the optimal clustering solution by themselves (see section 2.5.1 for further details).

Consequently, in order to avoid the disadvantages of this sub-optimal solution without giving up the advantages of the agglomerative hierarchical clustering methods, it would be more convenient to model learners' activity in online discussion forums by following an strategy based on an agglomerative hierarchical clustering algorithm that automatically determines the real number of clusters in the dataset, provides an entirely data-driven clustering solution and limits the user's subjective participation to the only stage of the process where is unavoidably required: the interpretation of the clustering results in terms of the learning behaviours performed by learners.

1.3 Thesis outline

Regarding the current status of the issue of modelling learners' activity in online discussion forums by means of cluster analysis, the following research questions arise:

Q1. May the real number of clusters and the final clustering solution of a dataset be automatically obtained by means of an agglomerative hierarchical clustering algorithm?

Q2. May this agglomerative hierarchical clustering algorithm deal with datasets of different nature that can contain clusters of distinct characteristics?

Q3. May this agglomerative hierarchical clustering algorithm automatically provide optimal clustering solutions without drastically increasing its computational requirements in comparison with other agglomerative hierarchical clustering algorithms?

Q4. May this agglomerative hierarchical clustering algorithm be employed to model learners' activity in online discussion forums limiting the user intervention to the interpretation of the final results?

In order to give a proper answer to these research questions, the present thesis poses the two following research hypotheses:

H1. Agglomerative hierarchical clustering methods are suitable for automatically determining the real number of clusters and providing the clustering solution on datasets of different nature that may contain clusters of distinct characteristics.

H2. Learners' participation in online discussion forums can be properly modelled and described by means of a clustering-based strategy that automatically provides the clustering solution and limits any user intervention to the interpretation stage of the analysis process.

The rest of the present thesis is structured as follows. An overview of clustering methods is presented in Chapter 2, followed by a more specific study on agglomerative hierarchical clustering algorithms performed in Chapter 3. Next, a novel parameter-free agglomerative hierarchical clustering algorithm is presented and described in Chapter 4. A first set of experimental results is shown in Chapter 5, where the performance of the new clustering algorithm is evaluated and compared with the capacities of other clustering algorithms. Once evaluated its general performance, an analysis strategy based on the new algorithm in order to model learners' activity in online discussion forums is built up in Chapter 6. And, finally, the conclusions of the thesis and the proposals for further work are presented in Chapter 7.

Chapter 2

Overview of clustering methods

As shown in the previous chapter, agglomerative hierarchical clustering methods are chosen to model learners' activity in online discussion forums when this issue is posed in terms of a clustering scenario. Nonetheless, the scope the clustering paradigm covers is huge and complex, since it involves a great diversity of principles, approaches, methods and techniques, and agglomerative hierarchical clustering methods are just a part of it. Quite obviously, the particularities of modelling learners' activity in online discussion forums via clustering techniques (mainly, being a context-dependent problem of exploratory nature and its high dependency on a proper estimation of the number of clusters, which is unknown) are the real motivations for this choice. However, the strategy for the estimation of the number of clusters carried out to date in this context seems to be an improvable solution, so that it remains unclear whether other clustering strategies may lead to better results. Thus, the goals of the present chapter are to provide an overview of the clustering paradigm, contextualising agglomerative hierarchical clustering methods within, and, as a first contribution of the present thesis, to survey the different approaches to the issue of the estimation of the number of clusters, analysing the benefits and drawbacks each one of them involve.

Hence, with no claim of being exhaustive, the different aspects relevant to the clustering paradigm are next studied, specially those of particular interest in the framework of this thesis. Thus, notational conventions are introduced in section 2.1; different proximity measures between objects are presented in section 2.2; a general categorisation of clustering methods is performed in section 2.3; diverse strategies to validate clustering results are examined in section 2.4; and different indeterminacies inherent to any clustering problem are studied in section 2.5, specially emphasising the problem of determining the number of clusters, which is studied in more detail. Finally, the reasons for the suitability of parameter-free agglomerative hierarchical clustering methods to the issue of modelling learners' activity in online discussion forums are summarised in section 2.6.

2.1 Notational conventions

Being widely employed throughout the present work, the following notational conventions are introduced:

- In accordance with the statements previously defined in section 1.2.3, a dataset $\mathbf{X}$ is constituted by a set of N objects represented upon a space of D features. That is:

$$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}, \forall N \in \mathbb{Z}^+ \tag{2.1}$$

  where $\mathbf{x}_i$ is the ith object in the dataset $\mathbf{X}$. On its behalf, $\mathbf{x}_i$ is denoted as a D-dimensional vector of features:

$$\mathbf{x}_i = [x_{i_1} x_{i_2} ... x_{i_D}], \forall D \in \mathbb{Z}^+ \tag{2.2}$$

  being x_{i_j} the jth feature of the ith object in dataset $\mathbf{X}$. Particularly, only real numerical features are considered in the present thesis, therefore $x_{i_j} \in \mathbb{R}$ and $\mathbf{x}_i \in \mathbb{R}^D$.

- As previously stated in section 1.2.3.2, the outcome of a clustering process is a set $\mathbf{P}$ of K clusters, such that:

$$\mathbf{P} = \{\mathbf{C}_1, \mathbf{C}_2, ..., \mathbf{C}_K\}, \forall K \in \mathbb{Z}^+ \tag{2.3}$$

  where $\mathbf{C}_i$ is the ith cluster in the set $\mathbf{P}$. In general terms, $\mathbf{C}_i$ can be defined as a subset of the dataset $\mathbf{X}$ ($\mathbf{C}_i \subseteq \mathbf{X}$), since it contains one or more objects belonging to $\mathbf{X}$:

$$\mathbf{C}_i = \left\{\mathbf{x}_{n_1^i}, \mathbf{x}_{n_2^i}, ..., \mathbf{x}_{n_{N_i}^i}\right\}, \forall N_i \leq N \mid N_i \in \mathbb{Z}^+ \tag{2.4}$$

  being n_j^i the index of the jth object belonging to cluster $\mathbf{C}_i$ and N_i the number of objects in cluster $\mathbf{C}_i$. Although any given couple of clusters $(\mathbf{C}_i, \mathbf{C}_j)$ belonging to $\mathbf{P}$ may easily be disjoint ($\mathbf{C}_i \cap \mathbf{C}_j = \emptyset$) or overlapped ($\mathbf{C}_i \cap \mathbf{C}_j \neq \emptyset$), every object of $\mathbf{X}$ belongs to at least one cluster in $\mathbf{P}$, therefore:

$$\bigcup_{i=1}^{K} \mathbf{C}_i = \mathbf{X} \tag{2.5}$$

2.2 Proximity measures

As previously stated in section 1.2.3.2, measuring the resemblance (or proximity) between the objects in the dataset is central to clustering processes. In general terms, there exist two complementary ways of comparing objects: measuring the distance (or dissimilarity) between them by means of a **distance function** or measuring their degree of similarity by means of a **similarity function** (Sevillano, 2009). For convenience, the term proximity is used in the present thesis to refer to either similarity or dissimilarity (Tan et al., 2006). In general terms, a proximity function on a dataset

$\mathbf{X}$ is defined such that it assigns a real value to every couple of objects in $\mathbf{X}$:

$$\begin{aligned} f : \mathbf{X} \times \mathbf{X} &\mapsto \mathbb{R} \\ \{\mathbf{x}_i, \mathbf{x}_j\} &\mapsto f(\mathbf{x}_i, \mathbf{x}_j), \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X} \end{aligned} \tag{2.6}$$

where $f(\mathbf{x}_i, \mathbf{x}_j)$ is a real value that indicates the proximity between $\mathbf{x}_i$ and $\mathbf{x}_j$.

The selection of the proximity measure is essentially problem-dependent (Xu and Wunsch II, 2005). A data object is described by a set of features, usually represented as a multidimensional vector (see section 2.1). The features can be quantitative or qualitative, continuous or binary, nominal or ordinal, which determine the corresponding measure mechanisms. Whilst distance functions are habitually used for measuring numerical features and similarity functions are more common for qualitative variables, there are no deterministic and systematic rules in order to decide what specific measure to apply (Jain et al., 1999; Xu and Wunsch II, 2005). Moreover, it may be added that there are multiple ways of transforming a similarity measure into a distance, and vice versa (Fenty, 2004).

Since only real numerical features are considered in the present thesis (see section 2.1), this section focuses on measures for computing the proximity between objects under numeric feature representations. A further insight into both the following and many other distance and similarity measures, their properties and other characteristics is provided by Duda et al. (2001), Xu and Wunsch II (2005) and Gan et al. (2007).

- **Distance functions**. Let $\mathbf{x}_i$ and $\mathbf{x}_j$ be two D-dimensional objects in the dataset $\mathbf{X}$. A distance function on $\mathbf{X}$ is denoted by $D(\mathbf{x}_i, \mathbf{x}_j)$ and is defined to satisfy the following two conditions (Xu and Wunsch II, 2005):

  1. Symmetry: $D(\mathbf{x}_i, \mathbf{x}_j) = D(\mathbf{x}_j, \mathbf{x}_i), \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$.
  2. Positivity: $D(\mathbf{x}_i, \mathbf{x}_j) \geq 0, \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$

  Furthermore, a distance function is called a metric if also holds the next two conditions:

  3. Triangle inequality: $D(\mathbf{x}_i, \mathbf{x}_j) \leq D(\mathbf{x}_i, \mathbf{x}_k) + D(\mathbf{x}_k, \mathbf{x}_j), \forall \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in \mathbf{X}$
  4. Reflexivity: $D(\mathbf{x}_i, \mathbf{x}_j) = 0 \Leftrightarrow \mathbf{x}_i = \mathbf{x}_j$

  Under these conditions, a wide set of distance functions are defined from the seminal **Minkowski distance**, which is actually a family of metrics (Gan et al., 2007):

  $$D(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^{D} |x_{i_k} - x_{j_k}|^p \right)^{\frac{1}{p}}, \forall p \in \mathbb{R}^+ \tag{2.7}$$

  Particular cases of Minkowski distance give rise to different distance functions, such as the **Manhattan (or City-block) distance** ($p = 1$), the **Maximum (or Sup) distance** ($p \to \infty$), or

the **Euclidean distance** ($p = 2$), which is the most commonly used numerical metric (Xu and Wunsch II, 2005):

$$D(\mathbf{x}_i, \mathbf{x}_j) = +\sqrt{\sum_{k=1}^{D} |x_{i_k} - x_{j_k}|^2} \tag{2.8}$$

- **Similarity functions**. Let $\mathbf{x}_i$ and $\mathbf{x}_j$ be two D-dimensional objects in the dataset $\mathbf{X}$. A similarity function on $\mathbf{X}$ is denoted by $S(\mathbf{x}_i, \mathbf{x}_j)$ and is defined to satisfy the following two conditions (Xu and Wunsch II, 2005):

  1. Symmetry: $S(\mathbf{x}_i, \mathbf{x}_j) = S(\mathbf{x}_j, \mathbf{x}_i), \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$.
  2. Positivity: $0 \leq S(\mathbf{x}_i, \mathbf{x}_j) \leq 1, \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}$

  Furthermore, a similarity function is called a similarity metric if also holds the next two conditions:

  3. $S(\mathbf{x}_i, \mathbf{x}_j) S(\mathbf{x}_j, \mathbf{x}_k) \leq [S(\mathbf{x}_i, \mathbf{x}_j) + S(\mathbf{x}_j, \mathbf{x}_k)] S(\mathbf{x}_i, \mathbf{x}_k), \forall \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in \mathbf{X}$
  4. $S(\mathbf{x}_i, \mathbf{x}_j) = 1 \Leftrightarrow \mathbf{x}_i = \mathbf{x}_j$

  Many similarity functions are defined under these conditions, such as the **Pearson correlation coefficient** or the **Extended Jaccard coefficient**, which measure similarity in terms of correlation between variables (Sevillano, 2009). One of the most commonly used is the **Cosine similarity**, which measures the angle comprised between the vectors representing the objects (Tan et al., 2006):

$$S(\mathbf{x}_i, \mathbf{x}_j) = \cos(\alpha) = \frac{\langle \mathbf{x}_i, \mathbf{x}_j \rangle}{\|\mathbf{x}_i\| \|\mathbf{x}_j\|} = \frac{\sum_{k=1}^{D} x_{i_k} x_{j_k}}{+\sqrt{\left(\sum_{l=1}^{D} x_{i_l}^2\right)\left(\sum_{m=1}^{D} x_{j_m}^2\right)}} \tag{2.9}$$

  where α is the angle between vectors $\mathbf{x}_i$ and $\mathbf{x}_j$. If x_{i_k} and/or x_{j_k} adopt negative values, the expression in equation 2.9 does not meet the positivity condition of a similarity function, since $-1 \leq \cos(\alpha) \leq 1$. Hence, it can be modified in order to fit to it, by defining $S(\mathbf{x}_i, \mathbf{x}_j) = \frac{1+\cos(\alpha)}{2}$. Furthermore, the **Cosine distance** can be derived from this similarity measure (Salton and Buckley, 1988):

$$D(\mathbf{x}_i, \mathbf{x}_j) = 1 - S(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{\sum_{k=1}^{D} x_{i_k} x_{j_k}}{+\sqrt{\left(\sum_{l=1}^{D} x_{i_l}^2\right)\left(\sum_{m=1}^{D} x_{j_m}^2\right)}} \tag{2.10}$$

Thus, considering that a proximity measure can be applied both between objects and between clusters (sets of objects) and since all the proximity measures used in the present thesis are based on distance functions, a simplified notation is employed from here on in order to refer to and differentiate between both kinds of proximities:

$$\begin{aligned} d_{\mathbf{x}_i \mathbf{x}_j} &= D(\mathbf{x}_i, \mathbf{x}_j) \\ d_{ij} &= D(\mathbf{C}_i, \mathbf{C}_j) \end{aligned} \tag{2.11}$$

Finally, all the possible pairwise proximities in a dataset $\mathbf{X}$ of N objects are contained in its proximity matrix (Jain and Dubes, 1988), which can be defined as:

$$M_P(\mathbf{X}) = \begin{pmatrix} 0 & d_{\mathbf{x}_1\mathbf{x}_2} & \dots & d_{\mathbf{x}_1\mathbf{x}_N} \\ d_{\mathbf{x}_2\mathbf{x}_1} & 0 & \dots & d_{\mathbf{x}_2\mathbf{x}_N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{\mathbf{x}_N\mathbf{x}_1} & d_{\mathbf{x}_N\mathbf{x}_2} & \dots & 0 \end{pmatrix} \tag{2.12}$$

where M_P is a symmetrical $N \times N$ matrix that contains $\frac{N^2-N}{2}$ different values (Gan et al., 2007).

2.3 Categorisation of clustering methods

The great diversity of existent clustering methods can be categorised according to two main criteria: the mapping between objects and clusters (or how the objects belong to the clusters) and the structure of the clustering solution (or how the clusters are related to each other). Thus, a two-dimensional frame of reference which allows categorising clustering algorithms in a broad sense –*i.e.* without resorting to their theoretical foundations– is defined (Sevillano, 2009).

Firstly, and regarding to how the objects are mapped onto the clusters, there exist two categories of clustering methods:

- **Hard (or crisp) clustering methods**, where objects' membership to clusters is binary; *i.e.* the object may belong (1) or not belong (0) to the cluster, so that an absolute belonging is established between objects and clusters (Jain et al., 1999). In a hard clustering solution, clusters may be either **exclusive** –every object belongs to one cluster and one cluster only– or **overlapped** –objects can simultaneously belong to more than one cluster– (Tan et al., 2006).
- **Soft (or fuzzy) clustering methods**, where objects' membership to clusters is established to a certain degree; *i.e.* the object belongs to the cluster with a membership weight that may adopt values between 0 (absolute non-belonging to the cluster) and 1 (absolute belonging to the cluster), so that a relative belonging is established between objects and clusters (Jain et al., 1999). In a soft clustering solution, clusters are fully-overlapped, since every object in the dataset is associated in some degree with every cluster (Tan et al., 2006).

Secondly, and regarding to the structure of clusters that form the clustering solution, two other categories of clustering methods are defined:

- **Partitional clustering methods**, where a one-layer structure of clusters (or a single partition of the data) is defined; *i.e.* all the clusters reside in the same level (Jain and Dubes, 1988).

  On the one hand, **hard partitional clustering** methods are most typically exclusive (from here on in the present thesis, "hard exclusive partitional clustering" will be referred as HPC),

giving therefore rise to a single partition of the data formed by non-overlapped clusters (Tan et al., 2006). Thus, having a dataset $\mathbf{X}$ of N objects and a HPC on $\mathbf{X}$ formed by a partition $\mathbf{P}$ of K clusters, the clustering solution is usually represented by a N-dimensional integer-valued label vector (Sevillano, 2009), such as:

$$\boldsymbol{\lambda} = [\lambda_1 \lambda_2 ... \lambda_N] \mid \lambda_i \in \{1, K\}, \forall i \in \{1, N\} \tag{2.13}$$

where cluster label λ_i indicates the cluster the ith object in $\mathbf{X}$ belongs to ($\lambda_i = j \Leftrightarrow \mathbf{x}_i \in \mathbf{C}_j$). It is worth noting that the symbolic nature of the cluster labels, as the same HPC solution may perfectly be represented by different versions of the same label vector resulting from any possible permutation of the cluster labels (Sevillano, 2009). For instance, the HPC solution on a toy dataset shown in Figure 2.2a can be represented by the label vector $\boldsymbol{\lambda} = [1\,1\,2\,2\,2\,2\,3\,3\,3]$, as well as by $\boldsymbol{\lambda} = [2\,2\,3\,3\,3\,3\,1\,1\,1]$ or $\boldsymbol{\lambda} = [3\,3\,2\,2\,2\,2\,1\,1\,1]$.

On the other hand, the clustering solution in **soft partitional clustering** (SPC) methods is formed by a single partition of fully-overlapped soft clusters and it is usually represented by a $N \times K$ real-valued clustering matrix $\boldsymbol{\Lambda}$, whose (i, j)th entry (Λ_{ij}) indicates the ith object's degree of membership to the jth cluster (Sevillano, 2009). Every membership weight in $\boldsymbol{\Lambda}$ is a real value between 0 and 1 ($\Lambda_{ij} \in \mathbb{R} \mid \Lambda_{ij} \in [0, 1]$) and the sum of weights for each object must equal 1 $\left(\sum_{j=1}^{K} \Lambda_{ij} = 1\right)$ (Tan et al., 2006). A SPC solution can be converted into a HPC solution ($\boldsymbol{\Lambda} \mapsto \boldsymbol{\lambda}$) by assigning each object to the cluster with the largest membership weight ($\Lambda_{ij} = \max\{\Lambda_{i1}, \Lambda_{i2}, ..., \Lambda_{iK}\} \rightarrow \lambda_i = j$) (Jain et al., 1999).

K-means (Forgy, 1965; MacQueen, 1967) and fuzzy c-means (Dunn, 1973; Bezdek, 1981) are the best-known and most widely used HPC and SPC algorithms, respectively.

- **Hierarchical clustering methods**, where a hierarchical structure of several layers of clusters (or several nested partitions of the data) is defined; *i.e.* clusters reside in different levels, in such a way that series of nested clusters (or subclusters) are defined (Jain and Dubes, 1988).

  Hard hierarchical clustering methods are by far the most common approach to hierarchical clustering (from here on in the present thesis, "hard hierarchical clustering" will be referred as HC) and, according to the way the hierarchical structure of clusters is constructed (see Figure 2.1), they are subdivided into **agglomerative hierarchical clustering** (AHC) methods, which perform a bottom-up process where clusters merge in and produce new clusters, and **divisive hierarchical clustering** (DHC) methods, which perform a top-down process where clusters split into subclusters (Everitt et al., 2011). Although literature provides some implementations of DHC algorithms such as MONA and DIANA (Kaufman and Rousseeuw, 1990) or DISMEA (MacQueen, 1967; Späth, 1980), AHC methods are the most widely used in practise (see Chapter 3 for further details), since DHC methods are more expensive in computational terms (Xu and Wunsch II, 2005).

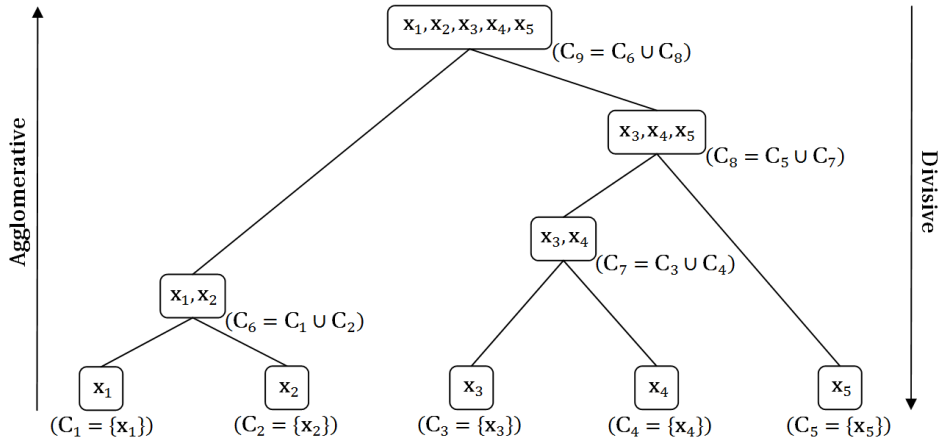
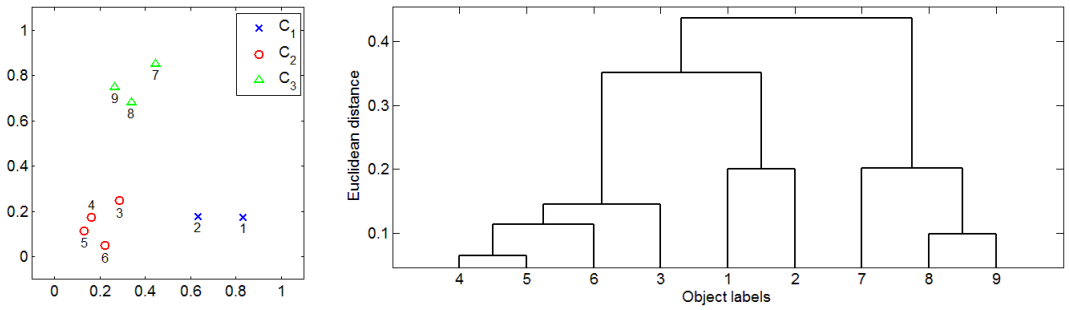


Figure 2.1: HC methods: agglomerative and divisive approaches (adapted from Gan et al. (2007)).

In HC methods, the clustering solution is defined by a hierarchy of clusters in a tree form that can be represented in many ways, being the dendrogram the most typical one (Gan et al., 2007). As shown in Figure 2.2b, a dendrogram is a branching diagram representing a hierarchical tree in which each internal node (or link) is associated to a cluster and indicates the distance between its two more immediate subclusters (Jain and Dubes, 1988).



(a) Scatterplot of the toy dataset.

(b) Dendrogram of the toy dataset.

Figure 2.2: An example of different clustering solutions on a 2-dimensional toy dataset. ***(a)*** The scatterplot shows the 9 objects of the dataset (each one identified by its object label) and a HPC solution of 3 clusters. ***(b)*** The dendrogram represents an AHC solution on the same dataset: from 9 singleton clusters on (each one contains one object), clusters merge in pairs forming new clusters (the links show the proximity between each pair of joined clusters) until a single final cluster is reached.

Thus, a HC solution on a dataset $\mathbf{X}$ of N objects is formed by a set $\mathbf{P}$ of $2N-1$ clusters, whose first N clusters are singleton ($\mathbf{C}_i = \{\mathbf{x}_i\}, \forall i \in \{1, N\}$) and whose last $N-1$ clusters are assigned a proximity value (Vandev and Tsvetanova, 1995):

$$\begin{aligned} &f : \mathbf{P} \mapsto \mathbb{R} \\ &\mathbf{C}_i \mapsto f(\mathbf{C}_j, \mathbf{C}_k) = d_i, \forall \mathbf{C}_i, \mathbf{C}_j, \mathbf{C}_k \in \mathbf{P} \mid i \in \{N{+}1, 2N{-}1\}, \mathbf{C}_i = \mathbf{C}_j \cup \mathbf{C}_k \end{aligned} \tag{2.14}$$

where d_i is the proximity level of cluster $\mathbf{C}_i$ and it is defined as the proximity between clusters $\mathbf{C}_j$ and $\mathbf{C}_k$ ($d_i = d_{jk}$). Hence, the dendrogram that comprises all the structure of the HC

solution can be represented as a $(N-1) \times 3$ real-valued matrix $\mathbf{\Delta}$, defined as:

$$\mathbf{\Delta} = \begin{pmatrix} \Delta_{11} & \Delta_{12} & d_{N+1} \\ \vdots & \vdots & \vdots \\ \Delta_{(N-1)1} & \Delta_{(N-1)2} & d_{(2N-1)} \end{pmatrix} \Leftrightarrow \mathbf{C}_{N+i} = \mathbf{C}_{\Delta_{i1}} \cup \mathbf{C}_{\Delta_{i2}} \tag{2.15}$$

where the ith row in $\mathbf{\Delta}$ $\left(\overline{\Delta}_i\right)$ represents the ith link in the dendrogram, which indicates that cluster $\mathbf{C}_{N+i}$ results from the union of clusters $\mathbf{C}_{\Delta_{i1}}$ and $\mathbf{C}_{\Delta_{i2}}$, whose proximity is $d_{(N+i)}$ (the proximity level of cluster $\mathbf{C}_{N+i}$). Dendrograms aside, Sneath and Sokal (1973) and Gan et al. (2007) can be consulted for other different representations of a HC solution.

Since HC can be viewed as a sequence of partitions of the data, different HPC solutions can be obtained from a single HC solution (Tan et al., 2006). In this sense, there are methods that postprocess dendrograms in order to obtain one or several label vectors (see section 3.1.3 for further details), or there even exist AHC algorithms that automatically yield to a hierarchically constructed HC solution where each cluster has its own particular dendrogram (see section 3.2 for further details).

Finally, much less usual are the **soft hierarchical clustering** (SHC) methods based of fuzzy set theories, which provide a clustering solution consisting of several layers of fully-overlapped hierarchically-structured soft partitions. Thus, a cluster belonging to a given layer may host a certain amount of clusters belonging to the immediately lower layer, and so recursively (Geva, 1999).

Next, in order to complete the categorisation of clustering methods performed in the context of the present thesis, a brief discussion focused on the advantages and drawbacks of AHC in comparison to HPC is presented in section 2.3.1 and the most common theoretical approaches clustering algorithms are based on are described in section 2.3.2.

2.3.1 AHC versus HPC

The present thesis focuses on the study of AHC. Thus, on the benefits side, when comparing them with HPC methods, AHC methods presents many advantages:

- AHC methods do not require the real number of clusters in the dataset to be known in advance (Kotsiantis and Pintelas, 2004). In fact, given the information about the structure of data provided by a dendrogram, AHC may certainly be a valid strategy in order to properly decide the value of K (Geva, 1999).
- Unlike many HPC methods (*e.g.* k-means algorithm), AHC methods set a deterministic initialisation –there are always N singleton clusters in the initial step–, so that the clustering solution provided by an AHC algorithm, and therefore its performance, does no depend on its initialisation (Tan et al., 2006).

- AHC methods are clearly more suitable than HPC methods to perform exploratory analysis of data, since they bring about useful visualisations of the clustering results (Kotsiantis and Pintelas, 2004). Since a dendrogram collects together many of the proximity and classificatory relationships in a body of data, it is therefore a convenient representation that allows to identify useful groups and salient interrelationships in the data (Murtagh and Contreras, 2012). Besides, due to both their flexibility regarding the level of granularity (Berkhin, 2006) and their ability to handle any attribute type, different proximity functions and different cluster sizes, densities and shapes (Tan et al., 2006), AHC methods are more versatile than HPC methods (Jain et al., 1999), since they are applicable in almost any clustering scenario, regardless of whether there are actually underlying hierarchical structures in data or not (Everitt et al., 2011).

- AHC methods compute a complete hierarchy of clusters and therefore provide a richer and more complex clustering solution, which itself includes many single partitions (HPC solutions) of the data –which partly justifies their higher computational cost– (Kotsiantis and Pintelas, 2004).

Nonetheless, on the drawbacks side, there are some criticisms commonly drawn by AHC methods:

- AHC methods lack a global objective function, which allows them to not have problems with local minima or difficulties in the initialisation stage, but makes them unable to optimise a global criterion (Tan et al., 2006).

- AHC methods are unable to perform adjustments once a merging decision is made (Kotsiantis and Pintelas, 2004). While HPC methods like k-means allow assignations between objects and clusters to vary throughout the iterations performed by the algorithm, merging decisions are final in AHC methods, since, once the decision of joining two clusters is taken, it cannot be undone at a later time (Tan et al., 2006). As a consequence, AHC methods tend to present a lower robustness against noise and outliers, since they are not capable of correcting possible misassignations (Xu and Wunsch II, 2005).

- AHC methods tend to have higher computational requirements, both in storage and time terms, than HPC methods (Tan et al., 2006). While the computational complexity of many HPC methods linearly grows with N –in asymptotic notation, their behaviour is $O(N)$ or $O(N \log N)$– (Xu and Wunsch II, 2005), most efficient implementations of AHC methods are $O\left(N^2\right)$ or $O\left(N^2 \log N\right)$ (Murtagh and Contreras, 2012), which made them much less suitable when dealing with very large datasets (Jain et al., 1999).

However, many new AHC techniques have been developed in recent years in order to improve the clustering performance and be able both to deal with large-scale datasets in spite of their computational requirements (Xu and Wunsch II, 2005) and to compensate their lack of flexibility once two

clusters are merged, either by constraining the agglomeration process using a previously obtained HPC solution –at the expense of requiring the value of K as an input parameter– (Zhao et al., 2005) or by improving the merging decision criteria used in the agglomeration process –in a free of input parameters approach– (see section 3.2 for further details).

2.3.2 Theoretical approaches to clustering

There exist a great diversity of clustering algorithms based on different foundations. A succinct description of the most well-known theoretical approaches for implementing clustering algorithms is next provided.

Centre-based clustering

Also known as squared error-based clustering, the goal of centre-based clustering is to minimise an objective function, which define how good the clustering solution is. In centre-based clustering algorithms, each cluster is represented by a centre (or a central element), which make them very efficient for clustering large and high-dimensional databases, but not very suitable to deal with clusters of arbitrary shapes –centre-based clustering algorithms tend to find convex-shaped clusters– (Gan et al., 2007). Based on the objective function defined by the sum of squared distances between objects and centroids, the k-means algorithm (Forgy, 1965; MacQueen, 1967) is probably the most representative example of this type of clustering (Sevillano, 2009). There also exist other centre-based HPC algorithms, like the ISODATA algorithm (Ball and Hall, 1965) and many variations of k-means developed to improve its performance and avoid some of its drawbacks, such as the k-medioids algorithm –which is more robust to outliers– (Kaufman and Rousseeuw, 1990) or the x-means algorithm –which automatically estimates the number of clusters– (Pelleg and Moore, 2000), among many others (Tan et al., 2006; Gan et al., 2007). SPC algorithms like fuzzy c-means (Dunn, 1973; Bezdek, 1981) and some of its variations –*e.g.* HUFC algorithm (Geva, 1999)– are based on optimising a global objective function as well. Finally, and despite their lack of a global objective function, some AHC methods can be seen as centred-based, since they either represent clusters by their geometric centres –Centroid (Sokal and Michener, 1958) and Median methods (Gower, 1967)–, well-scattered points –CURE algorithm (Guha et al., 1998)–, or height-balanced trees –BIRCH algorithm (Zhang et al., 1996)–, or use a merging criterion based on minimising a sum of squares error local objective function –Ward's method (Ward Jr., 1963; Ward Jr. and Hook, 1963)– (see section 3.1.2 for further details).

Graph-based clustering

Concepts and properties of graph theory can be connected to clustering (Hubert, 1974), since they may be applied to build a graph (or hypergraph) whose nodes (or links) reflect the similarity bet-

ween the objects in the dataset (Sevillano, 2009). This graph is usually obtained from the proximity matrix and it can be partitioned in order to cluster the data (Gan et al., 2007). Some of the most widely used AHC methods are graph-based, since they define the proximity between clusters by establishing different linkage criteria –Single-Link (Florek et al., 1951; McQuitty, 1957; Sneath, 1957), Complete-Link (Johnson, 1967; Lance and Williams, 1967) and Average-Link (Sokal and Michener, 1958; Lance and Williams, 1966) methods– (see section 3.1.1 for further details). There also exist other implementations of graph-based AHC, such as Chameleon (Karypis et al., 1999) and ROCK (Guha et al., 2000) algorithms, as well as graph-based criteria to obtain a single data partition from an AHC solution by dismissing the most inconsistent edges on a minimum spanning tree (Zahn, 1971) –which may be perfectly applied to dismiss the most inconsistent links on a dendrogram (see 3.1.3 for further details)–. HPC algorithms like CACTUS (Ganti et al., 1999), CLICK (Sharan and Shamir, 2000) or the Dynamic System-based Clustering algorithm proposed by (Gibson et al., 2000) and improved by (Zhang et al., 2000) are graph-based as well. Finally, spectral clustering is a recently emerged branch of graph-based clustering whose methods have proved to be effective for data clustering, image segmentation, web-ranking analysis and dimension reduction (Kannan et al., 2004; von Luxburg, 2007).

Model-based clustering

Following a probabilistic perspective, model-based clustering is based on the assumption that data come from a mixture of probability distributions, each of which typically represents a different cluster (Gan et al., 2007). Finding the clusters according to this approach usually consists on assuming that their probability distributions fit some kind of parametric model (*e.g.* Gaussian mixture models) (Sevillano, 2009) and the estimation of the parameters of the underlying models is usually posed from a maximum likelihood approach (Jain et al., 1999). The most common strategy to model cluster probability densities involves Gaussian distributions, which give rise to the multiple variants of the HPC methods based on the EM algorithm (McLachlan and Krishnan, 1997; Meng and van Dyk, 1997; Martínez et al., 2005), first formulated by Dempster et al. (1977), as well as the different AHC methods based on Gaussian models defined by Fraley and Raftery (1998) or the Bayesian AHC algorithm based on evaluating marginal likelihoods of a multivariate Gaussian model posed by Heller and Ghahramani (2005). Other algorithms implemented under this approach are AutoClass –which poses the use of Poisson, Bernoulli and log-normal probability distributions– (Cheeseman and Stutz, 1996), Snob –which combines a mixture model with the minimum message length principle– (Wallace and Dowe, 1994), COOLCAT –based on minimising the entropy of categorical attributes arrangements– (Barbará et al., 2002), STUCCO –uses significant contrast-sets that meaningfully differ among distinct groups of attributes– (Bay and Pazzani, 1999) or the Fuzzy Maximum-Likelihood Estimation (FMLE) algorithms proposed by Gath and Geva (1989a) and Gath and Geva (1989b).

Density-based clustering

Defining clusters as dense regions of objects separated by low-density regions (Gan et al., 2007) allows density-based clustering methods discovering arbitrarily shaped clusters and providing a natural protection against outliers (Sevillano, 2009). Furthermore, important advantages of this methodology are that only one scan of the dataset is needed, noise can be effectively handled and the number of clusters to initialise the algorithm is not required (El-Sonbaty et al., 2004; Murtagh and Contreras, 2012), although it may not be the most suitable choice when dealing with high-dimensional datasets (Gan et al., 2007). Depending on the way density is computed, two main approaches to density-based clustering are defined (Sevillano, 2009): while algorithms like DBSCAN (Ester et al., 1996) compute density directly from the objects in the dataset, algorithms like DENCLUE (Hinneburg and Keim, 1998) define analytical models of density over the features space. Moreover, there exist algorithms that integrate density-based clustering principles with other approaches, like centre-based –BRIDGE algorithm (Dash et al., 2001)–, graph-based –CUBN algorithm (Wang and Wang, 2003), Shared Nearest Neighbours (SNN) (Ertöz et al., 2002, 2003)– or model-based clustering –DBCLASD algorithm (Xu et al., 1998)–.

Grid-based clustering

With the aim of handling large datasets, the main idea of grid-based clustering methods is to use a grid-like structure to split the data space into cells, separating the dense grid regions from the less dense ones, and look for groups of objects (Murtagh and Contreras, 2012). The major advantage of grid-based clustering methods is the significant reduction of their computational complexity when dealing with large-scale high-dimensional datasets (Gan et al., 2007). Thus, the most typical approach within this methods consists of performing the following steps (Grabusts and Borisov, 2002): creating a grid structure –*i.e.* partitioning the data space into a finite number of non-overlapping cells–, calculating the cell density for each cell, sorting cells according to their densities, identifying cluster centres and traversing of neighbour cells. Some of the most well-known implementations based on this approach are HC algorithms like STING (Wang et al., 1997), OptiGrid (Hinneburg and Keim, 1999) or GRIDCLUS (Schikuta, 1996), and HPC algorithms like GDILC (Zhao and Song, 2001) or WaveCluster (Sheikholeslami et al., 1998). Further grid-based clustering algorithms can be found in (Chang and Jin, 2002), (Park and Lee, 2004) and (Xu and Wunsch II, 2008).

Combinatorial search-based clustering

Many clustering algorithms may not be able to obtain the global optimal clustering solution that fits the dataset, since they find local optimal partitions of the data. The aim of combinatorial search-based clustering methods is to search the clustering solution space and find the best par-

tition of the data in global terms –*i.e.* clustering is considered as a combinatorial optimisation problem– (Gan et al., 2007). In this context, combinatorial search-based clustering algorithms are usually based on either deterministic search techniques or stochastic optimisation methods (Sevillano, 2009). Thus, whereas deterministic annealing is the most typical deterministic search technique applied to clustering (Hofmann and Buhmann, 1997), other popular approaches are based on evolutionary computation (or genetic algorithms) (Hall et al., 1999; Tseng and Yang, 2001), simulated annealing (Selim and Al-Sultan, 1991), Tabu search (Al-Sultan, 1995) and hybrid solutions (Chu and Roddick, 2000; Scott et al., 2001). A compelling insight into several algorithms specifically implemented under these premises is provided by Gan et al. (2007).

Subspace clustering

Subspace clustering deals with the typical problems of high-dimensional datasets –the proximity between any given pair of objects may become the same (Beyer et al., 1999) and different clusters may be embedded in different feature subspaces of the high-dimensional data (Agrawal et al., 1998)– by identifying both clusters within different subspaces in the data and their relevant associated features –*i.e.* the features that engender the subspaces that host the clusters– (Gan et al., 2007). Subspace clustering is often based on the application of both feature extraction –such as Principal Component Analysis (PCA) (Jolliffe, 1986) or Independent Component Analysis (ICA) (Hyvärinen et al., 2001), among others (Fodor, 2003)– and feature selection –by selecting the best features under some kind of criterion (Molina et al., 2002; Dy and Brodley, 2004)– techniques, with the aim of finding optimal representation spaces for the data (Gan et al., 2007). There exist many clustering algorithms implemented under this approach, being CLIQUE (Agrawal et al., 1998), PROCLUS (Aggarwal et al., 1999) and ORCLUS (Aggarwal and Yu, 2000) the most popular ones, among many others (Gan et al., 2007).

Kernel-based clustering

Unlike the principles subspace clustering is based on, kernel-based clustering methods aim to non-linearly transform the objects in the dataset into a higher-dimensional feature space in order to separate these objects linearly (Sevillano, 2009). Thus, an inner-product kernel is designed in order to avoid the time-consuming (and sometimes even unfeasible) process of explicitly mapping the objects of the dataset into the new higher-dimensional transformed space (Xu and Wunsch II, 2005). Either partitional or hierarchical clusters can be formed by Support Vector Clustering (SVC) (Ben-Hur et al., 2001), the most well-known implementation of this approach, which can also be extended to allow for fuzzy membership (Chiang and Hao, 2003). Kernel-based clustering provides many benefits, such as the ability both to handle arbitrarily-shaped clusters and to deal with noise and outliers (Xu and Wunsch II, 2005).

Neural networks-based clustering

The learning and modelling abilities of neural networks may be easily exploited in order to solve clustering problems (Sevillano, 2009). From the paradigm of the competitive learning (Bishop, 1995), Self-Organising Maps (SOM) (Kohonen, 1990) and Generalized Learning Vector Quantization (GLVQ) (Karayiannis et al., 1996) are the most popular implementations of neural networks-based HPC, whereas Adaptive Resonance Theory (ART) (Carpenter and Grossberg, 1987) encompasses a whole family of neural networks architectures that can be used for both SPC (Carpenter et al., 1991) and HC (Wunsch II et al., 1993) –*e.g.* PART algorithm, which implements a neural-network inspired subspace clustering (Cao and Wu, 2002)–.

Consensus clustering

With the aim of being robust to some of the inherent indeterminacies of clustering paradigm (see section 2.5 for further details), the consensus clustering strategy compiles into a cluster ensemble as many individual clustering solutions as possible (regardless of which different configurations and/or algorithms these clustering solutions have been obtained from) and, in a fully unsupervised mode, achieves a consensus clustering solution comparable to (or even better than) the best individual clustering solution available (Sevillano, 2009). The consensus clustering framework is defined as the problem of combining multiple partitions of a set of objects into a single consolidated clustering solution without accessing the features or algorithms that determined these partitions (Strehl, 2002). Several works in the literature consider this approach to clustering as a central or collateral matter (Strehl, 2002; Fred and Jain, 2003; Fern and Lin, 2008; Sevillano, 2009).

2.4 Evaluation of clustering results

Since clustering algorithms can easily perform in different ways depending on the decisions made in the preprocessing stage and the configuration of parameters that determine their behaviour, most clustering-based applications require, according Figure 1.1, some evaluation (or validation) of their results in the postprocessing stage (Halkidi et al., 2002a). This evaluation task is the main subject of clustering validity methods, which provide a measure of the quality (or validity) of a clustering solution –by a good quality clustering solution it is meant a solution reflecting well the true group structure of the data– (Sevillano, 2009). The aims of clustering validity methods are both determining whether the clustering solution is meaningful –a valid clustering solution cannot reasonably occur by chance or as an artifact of the clustering algorithm– and helping to its correct interpretation (Jain et al., 1999), although they can be also utilised to test the clustering tendency of the data –whether there is an absence of clustering structure in the data or not–, as well as to estimate how many clusters are actually present in the data (see section 2.5.1 for further details).

The validity measures (or indices) for evaluating the quality of a clustering solution are traditionally categorised into three main types of indices, which are described next (Tan et al., 2006):

- **External clustering validity indices**, which measure the degree of resemblance of the clustering solution to a predefined and allegedly correct external cluster structure, also known as ground truth. External validation is a supervised measure of clustering goodness, since it uses information not present in the data themselves, and it is only applicable when a correct clustering solution is *a priori* known.
- **Internal clustering validity indices**, which measure the degree of fitting between the description of the data provided by the clustering solution and the data themselves. Since it uses only information present in the data, internal validation is an unsupervised measure of clustering goodness.
- **Relative clustering validity indices**, which compare different clustering solutions with each other. Thus, relative validation is not actually a separate type of cluster evaluation measure, but a specific use of such measures. In essence, the purpose of relative validation is, given a set of clustering solutions, to define an (external or internal) evaluation criterion and choose the best clustering solution accordingly (Halkidi et al., 2002b).

Most commonly, the interpretation of the value of a validity measure is made in statistical terms, since real (or natural) clusters tend to reflect non-random structure in the data and such structures should generate unusually significant values of the validity index. Besides, cluster evaluation involves more than obtaining a numerical measure of validity; the value of a clustering validation index has to be properly interpreted and the significance of the obtained results needs to be assessed (Tan et al., 2006).

Hence, some clustering scenarios may easily require of combining clustering validity indices with other statistical methods and/or testing hypotheses in order to characterise clustering results (Halkidi et al., 2001). Furthermore, the computational requirements entailed by the calculation of such validity indices and statistical measures have to be also taken into consideration (Berkhin, 2006).

Clustering evaluation criteria can be used for validating both individual clusters and complete clustering solutions (sets of clusters). Moreover, different types of both external and internal validity indices can be distinguish depending on the nature –partitional, hierarchical or soft– of the clustering solution (Gan et al., 2007). Nonetheless, since soft and hierarchical clustering results can always be converted to hard and partitional outcomes, the evaluation of HPC solutions is probably the most common cluster validation procedure (Strehl, 2002).

Thus, whereas SPC and SHC evaluation lies beyond the scope of this work (approaches to soft clustering validity are reported by Pal and Bezdek (1995), Dave and Krishnapuram (1997), Geva (1999) and Hammah and Curran (2000)), the specific HPC and HC validity methods used in the

present thesis are described in next sections. In particular, the Consistency Index (CI) is used as external HPC validity index (see section 2.4.1); the Silhouette Coefficient $(\overline{S})$ is used as internal HPC validity index (see section 2.4.2), along with the Kruskal-Wallis statistical hypothesis test (see section 2.4.3); and, finally, the Cophenetic Correlation Coefficient $(CPCC)$ is used as internal HC validity index (see section 2.4.4). A further insight into clustering validity can be gained by referring to diverse works particularly focused on this specific matter (Dubes and Jain, 1979; Dubes, 1993; Gordon, 1998; Halkidi et al., 2001, 2002a,b; Denoeud et al., 2006; Rendón et al., 2011).

2.4.1 The Consistency Index

Let $\mathbf{X}$ be a dataset constituted by a set of N objects, let $\mathbf{\Phi}$ be a HPC solution on $\mathbf{X}$ constituted by $K_{\mathbf{\Phi}}$ clusters $\left(\mathbf{\Phi} = \left\{\mathbf{C}_1^{\mathbf{\Phi}}, ..., \mathbf{C}_{K_{\mathbf{\Phi}}}^{\mathbf{\Phi}}\right\}\right)$, represented by label vector $\boldsymbol{\varphi}$ and considered as ground truth (*i.e.* a true and reliable clustering solution on $\mathbf{X}$) and, finally, let $\mathbf{P}$ be a HPC solution on $\mathbf{X}$ provided by

Algorithm 1 Calculation of the Consistency Index (adapted from Fred (2001)).

Input: HPC solutions $\mathbf{\Phi}$ (ground truth) and $\mathbf{P}$

procedure

$X_{\mathbf{\Phi}} : X_{\mathbf{\Phi}}(i,j) \leftarrow \begin{cases} 1 \; if \; \mathbf{x}_i \in \mathbf{C}_j^{\mathbf{\Phi}} \\ 0 \; otherwise \end{cases}, \forall \mathbf{x}_i \in \mathbf{X}, \forall \mathbf{C}_j^{\mathbf{\Phi}} \in \mathbf{\Phi}$

$X_{\mathbf{P}} : X_{\mathbf{P}}(i,j) \leftarrow \begin{cases} 1 \; if \; \mathbf{x}_i \in \mathbf{C}_j^{\mathbf{P}} \\ 0 \; otherwise \end{cases}, \forall \mathbf{x}_i \in \mathbf{X}, \forall \mathbf{C}_j^{\mathbf{P}} \in \mathbf{P}$

$N_{\mathbf{\Phi}} : N_{\mathbf{\Phi}}(i) \leftarrow \sum_{j=1}^{N} X_{\mathbf{\Phi}}(j,i), \forall i \in \{1, K_{\mathbf{\Phi}}\}, \forall j \in \{1, N\}$

$N_{\mathbf{P}} : N_{\mathbf{P}}(i) \leftarrow \sum_{j=1}^{N} X_{\mathbf{P}}(j,i), \forall i \in \{1, K_{\mathbf{P}}\}, \forall j \in \{1, N\}$

$M_{\mathbf{P}}^{\mathbf{\Phi}} \leftarrow (X_{\mathbf{\Phi}})^T \cdot X_{\mathbf{P}}$ ▷ ^{T} denotes transposition

$cross \leftarrow M_{\mathbf{P}}^{\mathbf{\Phi}}$

$N_{shared} \leftarrow 0$

$map_{\mathbf{P}}^{\mathbf{\Phi}} : map_{\mathbf{P}}^{\mathbf{\Phi}}(i) \leftarrow 0, \forall i \in \{1, K_{\mathbf{\Phi}}\}$

for $n \leftarrow 1, \min\{K_{\mathbf{\Phi}}, K_{\mathbf{P}}\}$ **do**

  $(k,l) \leftarrow \arg\max_{(i,j)} \left\{ \frac{cross(i,j)}{N_{\mathbf{\Phi}}(i) + N_{\mathbf{P}}(j) - M_{\mathbf{P}}^{\mathbf{\Phi}}(i,j)} \right\}$

  $N_{shared} \leftarrow N_{shared} + M_{\mathbf{P}}^{\mathbf{\Phi}}(k,l)$

  $map_{\mathbf{P}}^{\mathbf{\Phi}}(k) \leftarrow l$

  $cross(k,i) \leftarrow 0, \forall i \in \{1, K_{\mathbf{P}}\}$

  $cross(i,l) \leftarrow 0, \forall i \in \{1, K_{\mathbf{\Phi}}\}$

end for

$CI \leftarrow \frac{N_{shared}}{N}$

end procedure

Output: Consistency Index CI, matching matrix $M_{\mathbf{P}}^{\mathbf{\Phi}}$ and mapping of cluster labels $map_{\mathbf{P}}^{\mathbf{\Phi}}$

any given clustering algorithm, constituted by $K_{\mathbf{P}}$ clusters $\left(\mathbf{P} = \left\{\mathbf{C}_1^{\mathbf{P}}, ..., \mathbf{C}_{K_{\mathbf{P}}}^{\mathbf{P}}\right\}\right)$ and represented by label vector $\boldsymbol{\lambda}$.

The Consistency Index (CI) provides an external measure of the validity of $\mathbf{P}$ as HPC solution on $\mathbf{X}$, it is defined as the fraction of shared objects in matching $\boldsymbol{\lambda}$ with $\boldsymbol{\varphi}$ over N (Fred, 2001) and it is calculated as shown in Algorithm 1.

The CI can be considered as an unsupervised measure of accuracy. Its range of values is $[0, 1]$, where high values indicate great accuracy of the obtained HPC solution –great resemblance to the ground truth–. Its calculation provides with a mapping $\left(map_{\mathbf{P}}^{\boldsymbol{\Phi}}\right)$ between the cluster labels of $\boldsymbol{\lambda}$ and $\boldsymbol{\varphi}$, and the matching matrix –an unsupervised version of the confusion matrix– between $\mathbf{P}$ and the ground truth $\left(M_{\mathbf{P}}^{\boldsymbol{\Phi}}\right)$, which, combined with $map_{\mathbf{P}}^{\boldsymbol{\Phi}}$, allows to gain an insight into the differences between $\boldsymbol{\lambda}$ and $\boldsymbol{\varphi}$. Finally, it can also be applied to determine how similar two any given partitions on the same dataset are (Fred, 2001).

2.4.2 The Silhouette Coefficient

Let $\mathbf{X}$ be a dataset constituted by a set of N objects and let $\mathbf{P}$ be a HPC solution on $\mathbf{X}$ constituted by K clusters $(\mathbf{P} = \{\mathbf{C}_1, ..., \mathbf{C}_K\})$ and represented by label vector $\boldsymbol{\lambda}$.

The Silhouette Coefficient ($\overline{S}$) provides an internal measure of the validity of $\mathbf{P}$ as HPC solution on $\mathbf{X}$, it combines both cluster cohesion (or compactness) and cluster separation (or isolation) concepts (Tan et al., 2006), and it is defined as (Rousseeuw, 1987):

$$\overline{S} = \sum_{i=1}^{N} s\left(i\right) \tag{2.16}$$

where $s\left(i\right)$ is the individual silhouette coefficient of $\mathbf{x}_i$ –the ith object in $\mathbf{X}$– and it is defined as:

$$s\left(i\right) = \frac{b\left(i\right) - a\left(i\right)}{\max\left\{a\left(i\right), b\left(i\right)\right\}} \tag{2.17}$$

The term $a\left(i\right)$ is the average proximity of $\mathbf{x}_i$, which belongs to cluster $\mathbf{C}_j$, to the objects also belonging to cluster $\mathbf{C}_j$:

$$a\left(i\right) = \frac{1}{N_j - 1} \sum_{\substack{l=1 \\ l \neq i}}^{N_j} d_{\mathbf{x}_i \mathbf{x}_l}, \forall \mathbf{x}_l \in \mathbf{C}_j \tag{2.18}$$

being N_j the number of objects belonging to cluster $\mathbf{C}_j$; the term $b\left(i\right)$ is defined as:

$$b\left(i\right) = \min_{k \neq j}\left\{b_k\left(i\right)\right\}, \forall k \in \{1, K\} \tag{2.19}$$

where $b_k\left(i\right)$ is the average proximity of $\mathbf{x}_i$ to the objects belonging to cluster $\mathbf{C}_k$:

$$b_k\left(i\right) = \frac{1}{N_k} \sum_{l=1}^{N_k} d_{\mathbf{x}_i \mathbf{x}_l}, \forall \mathbf{x}_l \in \mathbf{C}_k \tag{2.20}$$

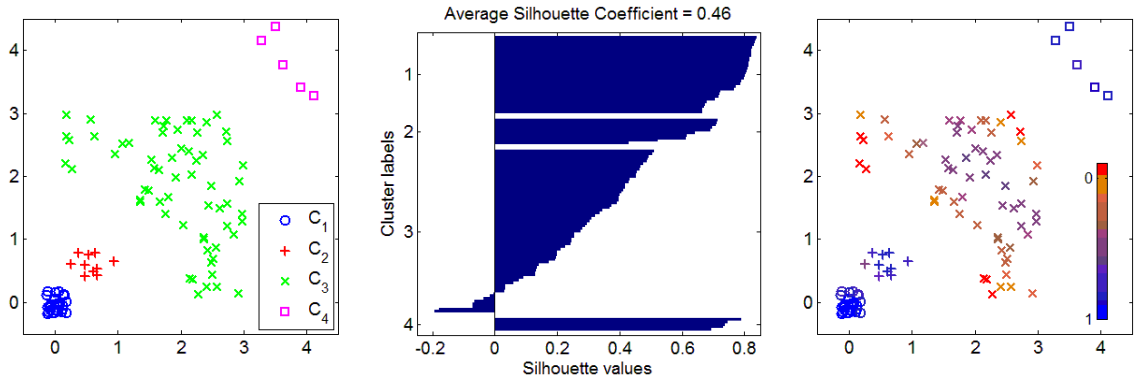
being N_k is the number of objects belonging to cluster $\mathbf{C}_k$.

If the set of individual silhouette coefficients associated to the objects belonging to a same cluster is considered, the average of the $a(i)$ values can be seen as a cohesion measure of the cluster –the lower the average value, the higher its cohesion–, whereas the average of the $b(i)$ values can be seen as an isolation measure of the cluster –the higher the average value, the higher its separation degree with respect to the rest of the clusters– (Tan et al., 2006).

The Silhouette Coefficient may indicate the quality of a HPC solution. The range of values of $\overline{S}$ –as well as of each $s(i)$– is $[-1, 1]$, where high values can be interpreted as a result of a set of high-compacted and high-isolated clusters, which are assumed by this validity measure to be the characteristics of a high-quality HPC solution (Rousseeuw, 1987).

Nonetheless, it may easily occur that differently-compacted and/or differently-shaped real clusters in the dataset are not suited for the cohesion and/or isolation measures implemented by this validity index. Hence, the Silhouette Coefficient, as every internal validity index, has its issues and limitations and they have to be considered in order to make a proper interpretation of its value (Halkidi et al., 2002a,b).

To that effect, let's consider the 2-dimensional toy dataset shown in Figure 2.3a (from here on in the present thesis, it will be referred as the *4toy* dataset). Being a synthetic dataset, its ground truth solution is known *a priori*, it is the best HPC solution on this dataset $(CI = 1)$ and it shows the four different clusters the toy dataset is actually composed of. As shown in Figures 2.3b and 2.3c, objects in clusters $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_4$ have high silhouette coefficient values, whereas objects in $\mathbf{C}_3$ present low –even negative– silhouette coefficient values.



(a) Scatterplot of the *4toy* dataset. ***(b)*** Silhouette of the ground truth. ***(c)*** Scatterplot of the silhouette.

Figure 2.3: Performance of the Silhouette Coefficient on the *4toy* dataset. ***(a)*** A 2-dimensional toy dataset composed of four unbalanced non-overlapped differently-compacted differently-shaped real clusters. Every object in the dataset is coloured according to its cluster label in the ground truth solution. ***(b)*** The silhouette of the ground truth solution plots the value of every individual silhouette coefficient $(s(i))$ –the whole silhouette of a HPC solution can be a useful graphical aid to interpret and validate clustering results–. The average Silhouette Coefficient $(\overline{S} = 0.46)$ is indicated as well. ***(c)*** Every object in the dataset is coloured in the scatterplot of the silhouette according to its individual silhouette coefficient; the more blue the colour, the higher the value, whereas red colours indicate poor values $(s(i) < 0)$.

Thus, this example illustrates how this cluster validity index performs poorly when dealing with low-compacted and/or arbitrarily-shaped clusters ($\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_4$ are much more compact than $\mathbf{C}_3$ and therefore their silhouette coefficients are clearly higher). Furthermore, the value of the average Silhouette Coefficient $\left(\overline{S} = 0.46\right)$ is far from its maximum $\left(\overline{S} \in [-1, 1]\right)$ even for a ground truth solution, which indicates that good HPC solutions may not have high values of $\overline{S}$ (positive, in general, but not necessarily high).

2.4.3 Kruskal-Wallis statistical hypothesis test

Statistical hypothesis testing provides a formal way to decide if the results of an experiment are significant or accidental. In standard statistical terminology, a set of n measurements (or observations) on a given variable (or dimension) d is called a sample S on d of size n. Let H_0 be the null hypothesis that assume that all the observations in the sample S come from a given distribution D_0. A statistical hypothesis test allows to determine whether the null hypothesis can be rejected; that is, to state with some degree of confidence –usually expressed as a probability– that all the observations in the sample S do not come from the same distribution (Duda et al., 2001).

In a clustering scenario, statistical hypothesis tests can be useful as a complement for the clustering validity indices with the aims of determining the quality of the clustering solution and achieving a better characterisation of the obtained clusters (Wise et al., 2013). Taking a dataset of objects as a sample of measurements, the features as variables and the clusters as groups of measurements belonging to the same sample, if a statistical hypothesis test on the objects belonging to different clusters along a feature d results in a rejection of the null hypothesis, statistically significant differences between clusters along the feature d can be assumed, since objects belonging to different clusters would actually come from different distributions –which can be seen as an indicator of a quality clustering solution inasmuch as the origin of the clusters is not arbitrary– (del Valle and Duffy, 2009; Khan et al., 2012).

In this context, the Kruskal-Wallis statistical hypothesis test is a non-parametric one-way method for the analysis of variance by ranks proposed by Kruskal and Wallis (1952). It can be used for comparing two or more groups of measurements belonging to the same sample and it allows to determine whether these groups are originated from the same distribution. Unlike its equivalent parametric one-way method (the ANOVA test), the Kruskal-Wallis test does not assume any specific distribution for the origin of the sample in the null hypothesis (the ANOVA test assumes a normal distribution), although it assumes identically shaped and scaled distributions for each group. The Kruskal-Wallis test can deal with unbalanced groups –of unequal size– and it is an extension of the Mann-Whitney test for three or more groups –both methods are the same when comparing two groups– (Conover, 1980).

Hence, being $\mathbf{X}$ a dataset constituted by a set of N D-dimensional objects and being $\mathbf{P}$ a HPC

solution on $\mathbf{X}$ formed by K clusters ($\mathbf{P} = \{\mathbf{C}_1, ..., \mathbf{C}_K\}$), the Kruskal-Wallis statistical hypothesis test consists of ranking all objects in the dataset from 1 to N considering the values of their ith feature (tied objects –*i.e.* objects equally ranked– are assigned to the mean of the ranks for which they are tied) and then computing the following H statistic (Kruskal and Wallis, 1952):

$$H = 3\,(N-1)\left(4\sum_{c=1}^{K}\frac{R_c^2}{n_c} - N(N+1)^2\right)\left(N\left(N^2-1\right) - \sum_{j=1}^{G} t_j^3 - t_j\right)^{-1} \tag{2.21}$$

where n_c is the number of objects belonging to cluster c, R_c is sum of the ranks of the objects belonging to cluster c, G is the number of groupings of objects that are tied at a particular value and t_j is the number of tied objects in the jth grouping.

Finally, the result of the test is given by the p-value of the H statistic ($P\left(\chi^2 \geq H\right)$, usually referred as p), which is the significance level of the test under the null hypothesis given by the χ^2 statistic approximation and which can be interpreted as the probability of the null hypothesis being true given the obtained value of H. This p-value can be obtained from the χ^2 approximation of the distribution of the H statistic under the null hypothesis provided by Kruskal and Wallis (1952).

Thus, obtaining a p-value less than the predetermined significance level –typically, $p < 0.01$ (Duda et al., 2001)– allows to reject the null hypothesis. Since the null hypothesis in the Kruskal-Wallis test is that objects belonging to different clusters come from the same population, if $p < 0.01$, objects belonging to two or more different clusters are assumed to come from different populations, therefore there exist significant differences among, at least, two clusters in the dataset along the ith feature (del Valle and Duffy, 2009; Khan et al., 2012; Wise et al., 2013).

A further insight into non-parametric statistical hypothesis testing can be gained by referring to Conover (1980), Siegel and Castellan Jr. (1988), Spurrier (2003) and Fay and Proschan (2010).

2.4.4 The Cophenetic Correlation Coefficient

Let $\mathbf{X}$ be a dataset constituted by a set of N objects and let $\mathbf{P}$ be a HC solution on $\mathbf{X}$ constituted by $2N-1$ clusters $\left(\mathbf{P} = \left\{\mathbf{C}_1, ..., \mathbf{C}_{(2N-1)}\right\}\right)$ and represented by dendrogram $\boldsymbol{\Delta}$.

The Cophenetic Correlation Coefficient ($CPCC$) provides an internal measure of the validity of $\mathbf{P}$ as HC solution on $\mathbf{X}$ by means of measuring the degree of similarity between the so-called cophenetic matrix of the HC solution (M_C) and the proximity matrix of the data (M_P). M_C is a symmetrical $N \times N$ matrix, containing $\frac{N^2-N}{2}$ different values and defined in such a way that its (i,j)th entry is the proximity level at which objects $\mathbf{x}_i$ and $\mathbf{x}_j$ are found in the same cluster for the first time (in an agglomerative sense) in the dendrogram $\boldsymbol{\Delta}$ (*i.e.*, the proximity level of the smallest cluster in the dendrogram that contains objects $\mathbf{x}_i$ and $\mathbf{x}_j$). Hence, $CPCC$ is the result of comparing M_P

and M_C and it is defined as (Sokal and Rohlf, 1962):

$$CPCC = \frac{\frac{1}{M}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} d_{ij}c_{ij} - \mu_P\mu_C}{\left(\frac{1}{M}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} d_{ij}^2 - \mu_P^2\right)\left(\frac{1}{M}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} c_{ij}^2 - \mu_C^2\right)} \tag{2.22}$$

where $M = \frac{N^2-N}{2}$; d_{ij} and c_{ij} are the (i,j)th entries of matrices M_P and M_C, respectively; and μ_P and μ_C are the average values of the elements in M_P and M_C, respectively:

$$\mu_P = \frac{1}{M}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} d_{ij} \quad , \quad \mu_C = \frac{1}{M}\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} c_{ij} \tag{2.23}$$

The Cophenetic Correlation Coefficient establishes the validity of a HC solution by determining how faithfully the dendrogram preserves the pairwise proximities between objects represented in M_P (M_C can be considered as a sort of proximity matrix defined from the dendrogram, which is in its turn built from M_P). The range of values of $CPCC$ is $[-1,1]$, where high values indicate great similarity between M_P and M_C and, hence, can be interpreted as indicators of a quality HC solution (Gan et al., 2007).

Nonetheless, high values of $CPCC$ require of a hierarchically structured dataset which gives rise to a M_P formed by hierarchically structured pairwise proximities, since pairwise proximities in M_C are derived from the dendrogram and therefore will always be hierarchically structured. This fact leads to a limitation of the $CPCC$ as an internal clustering validity index (see section 3.1.3 for further details), since it has been well proven that HC methods can be perfectly applied regardless of whether the data actually fit a hierarchical structure or not (Jain et al., 1999; Everitt et al., 2011).

Examples in section 3.1.3 show that $CPCC$ is a relatively reasonable indicator regarding whether real hard partitional clusters are properly contained within the hierarchical structure of the dendrogram. Although high values of $CPCC$ (close to 1) should not be expected unless the data are actually organised under some hierarchical structure, low values of $CPCC$ (close to 0 or negative) can be easily interpreted as indicators of a low-quality HC solution in almost any clustering scenario.

Finally, and in addition to these limitations, it is worth noting that tests for the absence of cluster structure and procedures for validating both partitions and individual clusters can also be applied when evaluating HC solutions (Gordon, 1996). Probably for these reasons, the most common approach to validation of hierarchies consists in validating partitions, since hierarchies can be seen as successions of partitions and different HPC solutions can be obtained from a single dendrogram (Sousa and Tendeiro, 2005).

2.5 Clustering indeterminacies

In accordance with the impossibility theorem for clustering stated by Kleinberg (2002) –whilst tinged by Carlsson and Mémoli (2009) in the case of some specific AHC methods–, clustering is a variable, complex and context-dependent task, hard to be tackled in its entirety from an unified framework. Thus, like any other task belonging to the KDD paradigm, a clustering process requires making several critical decisions at each of its stages (Sevillano, 2009). Besides, due to the unsupervised nature of clustering, these decisions are often made blindly and may determine to a large extent the effectiveness and suitability of the clustering results (Jain et al., 1999).

Hence, obtaining a quality clustering solution is closely related with and strongly dependent on making optimal decisions at every stage of the KDD process (see Figure 1.1). Sevillano (2009) refers to the uncertainties inherent to any clustering process as clustering indeterminacies and states two main types of sources of indeterminacy: indeterminacies at the preprocessing stage (the way objects are characterised in the dataset) and indeterminacies at the clustering stage (selection and configuration of the clustering algorithm).

Firstly, regarding decisions on data characterisation, optimal features would allow to easily distinguish –by means of using a proper proximity function– among objects belonging to different clusters, would be robust to noise and should provide meaningful interpretations of the data (Xu and Wunsch II, 2005). The main questions about this source of indeterminacy might be the following (Sevillano, 2009):

- How should the objects in the dataset be represented? By their original representation, by selecting a subset of the original features (*i.e.* feature selection) or by transforming original features into new ones (*i.e.* feature extraction)?
- Should the original data be subjected to a dimensionality reduction process? Which should be the dimensionality of the reduced feature space?
- In case that a feature selection/extraction process is applied, which selection/extraction criteria should be followed?
- How should objects be compared? By means of a distance function or a similarity function? Which specific proximity function should be used?

And secondly, the main questions about the selection and configuration of the particular clustering algorithm to apply might be the following (Sevillano, 2009):

- What type of clustering method should be applied? Hard or soft? Hierarchical or partitional?
- What type of theoretical approach to clustering should be considered?

- Once method and approach are selected, which clustering algorithm should be applied?
- How should the configuration parameters of the clustering algorithm be tuned, if any?
- How many clusters should the clustering algorithm reveal?

In addition to these proposals, a third source of indeterminacy might be considered depending on the scenario: indeterminacies relating to the clustering assessment stage (*i.e.* what cluster validity index/indices should be selected to evaluate the clustering results?). In any case, these indeterminacies are hard to handle in general terms due to the context-dependent nature of clustering problems. An interesting and worthy discussion about these issues is performed in Sevillano (2009, Section 1.4, pages 21–26), where multiples references to works in literature specifically focused on these matters are provided. Milligan (1996, Section 2, pages 341–343) and Everitt et al. (2011, Section 9.2, pages 260–262) can be also consulted in order to complement and complete this discussion.

From amongst these indeterminacies belonging to the clustering paradigm and its application in real-life scenarios, the present thesis focuses on the problem of determining the number of clusters, which has a great influence on the quality of the clustering solution (Xu and Wunsch II, 2005) and which has been pointed as the fundamental problem of cluster validity (Dubes, 1993).

2.5.1 How many clusters?

The problem of deciding the number of clusters (K) better fitting a dataset is one of the major issues of the clustering paradigm and it has been subject of many research efforts (Milligan and Cooper, 1985; Dubes, 1987; Gath and Geva, 1989b; Dave, 1996; Rezaee et al., 1998; Dimitriadou et al., 2002). Most clustering algorithms ask K to be provided as an input parameter, being quite obvious that the quality of resulting clustering solution is largely dependent on the estimation of K. Although users may be able to determine K according to their expertise in some particular scenarios, the value of K is unknown under generic circumstances and it should be estimated exclusively from the data themselves (Xu and Wunsch II, 2005).

The issue of how to properly estimate the optimal value of K has been tackled from the literature by adopting a great diversity of strategies, which are next surveyed. Thus, as a first contribution of the present thesis, a conceptual taxonomy is built throughout the following sections in order to contextualise such strategies according to the approach they adopt.

2.5.1.1 Exploratory approach

An exploratory analysis of the dataset prior to the execution of the clustering algorithm may provide useful notions to decide the value of K (Tukey, 1977). Such exploration may involve a direct

observation of the data –by means of visualisation techniques such as scatterplots, histograms, parallel coordinates, tree maps, etc.– (Gan et al., 2007, Chapter 5, pages 53–65), the use of statistical summarisation techniques –usually based on the calculation of statistics such as mean, median, percentiles, mean, standard deviation, covariance, skewness, kurtosis, etc.– (Tan et al., 2006, Section 3.2, pages 98–105), or the projection (and posterior visualisation) of the data into optimal low-dimensional representation spaces by means of feature selection/extraction techniques (Fodor, 2003; Molina et al., 2002; Dy and Brodley, 2004).

The fact that the final selection of the value of K comes under user's subjective criterion is the main drawback of this approach, which can easily lead to a non-optimal estimation of K. Moreover, the complexity of most real datasets also restricts the effectiveness of this approach only to a small scope of scenarios (Xu and Wunsch II, 2005).

2.5.1.2 Heuristic-based approach

In case of having some previous knowledge about the dataset and its characteristics, heuristic approaches based on a variety of techniques and theories may be followed. Depending on both the particular nature of the data and the kind of clustering method, literature on the specific area may provide different strategies, such as, by way of example, the eigenvalue decomposition of the feature space as an indicator of the possible existence of clusters proposed by Girolami (2002) in kernel-based clustering, the method based on the proximities between neighbour centroids developed by Kothari and Pitts (1999) for the k-means algorithm, the estimation of K based on influence zones proposed by Herbin et al. (2001) in image segmentation problems, or the dynamic branch cutting criterion defined by Langfelder et al. (2008) for HC of genomic data.

The application of such approaches is often subject to both specific datasets and particular clustering algorithms. Being, therefore, hardly generalisable, this approach usually lacks robustness and it may easily lead to non-optimal estimations of K (Xu and Wunsch II, 2005).

2.5.1.3 Relative validity approach

One of the most common approaches to the issue of determining K is based on the comparison of different clustering solutions; that is, on the relative validation (see section 2.4) of a set of different clustering solutions generated by a given clustering algorithm, where the selected value of K is that which belongs to the best clustering solution of the set according to some (habitually internal) validity index (Tan et al., 2006). Depending on the nature of the validity index, the best clustering solution may either optimise the index value (Milligan and Cooper, 1985) or give rise to a distinct knee, peak or inflection point in the plot of the obtained validity index values against the number of clusters of their associated clustering solutions (Tan et al., 2006, Section 8.5.5, pages 546–547).

Let's consider, by way of example, the *4toy* dataset defined in Figure 2.3a. After several executions of the k-means algorithm on this dataset –Euclidean distance has been used for comparing objects, the value of K has been varied for each execution ($K \in \{1, 20\}$) and centroids have been randomly initialised–, two different internal validity indices are utilised to evaluate every obtained clustering solution: the Silhouette Coefficient (see section 2.4.2) and the mean squared error (MSE), which is defined as (Tan et al., 2006):

$$MSE = \sum_{i=1}^{K} \frac{1}{N_i} \sum_{j=1}^{N} w_{ji}\, d_{\mathbf{x}_j \mathbf{c}_i} \tag{2.24}$$

where $\mathbf{c}_i$ is the centroid of cluster $\mathbf{C}_i$, N_i is the number of objects in cluster $\mathbf{C}_i$, $d_{\mathbf{x}_j \mathbf{c}_i}$ is the proximity between $\mathbf{x}_j$ and $\mathbf{c}_i$, and w_{ji} is a weight factor that equals 1 if $\mathbf{x}_j$ belongs to cluster $\mathbf{C}_i$ and 0 otherwise. The different values obtained for both validity indices are plotted against the number of clusters of their associated clustering solutions in Figures 2.4a and 2.4b, respectively. Both validity criterion suggest the presence of an optimal clustering solution at $K = 5$ (which is shown in Figure 2.4c): the Silhouette Coefficient is objectively maximised at $K = 5$, whereas a singular inflection point (note the distinctively stressed decreasing of the function between $K = 4$ and $K = 5$) might be manually identified by user in the MSE plot at $K = 5$ as well.

(a) $\overline{S}$ versus K. ***(b)*** *MSE* versus K. ***(c)*** Selected clustering solution.

Figure 2.4: Relative validity criteria for the estimation of the number of clusters (K) on the *4toy* dataset. ***(a)*** Silhouette Coefficient ($\overline{S}$) values plotted against the number of clusters of the generated clustering solutions. A maximum value is reached at $K = 5$. ***(b)*** MSE values plotted against the number of clusters of the generated clustering solutions. A distinctive inflection point might be identified at $K = 5$. ***(c)*** Scatterplot of the clustering solution with $K = 5$.

It is worth noting that the clustering solution that optimises both criteria is not the optimal clustering solution on the *4toy* dataset, which leads to a poor estimation of the real number of clusters –its ground truth solution contains 4 clusters (see Figure 2.3a)–. This is due to the fact that k-means algorithm fails to obtain a good clustering solution for $K = 4$ (probably because of a poor initialisation of the centroids) and it illustrates the limitations of estimating the value of K from this approach.

Thus, despite being widely used, this approach presents several drawbacks that may cause a non-optimal estimation of the value of K. Firstly, its main lack lies in the fact that the set of evaluated

clustering solutions must include the optimal one (or, at least, it must include a quasi-optimal solution of the same number of clusters as the optimal one), since the choice of K is subject to the selection of the best clustering solution within the set. Hence, as shown in the example of Figure 2.4, if none of the solutions in the set is at least close to the optimal one, the value of K can be easily miscalculated. Secondly, the necessity of obtaining a diversity of clustering solutions increments the computational cost of the process (Everitt et al., 2011). Thirdly, a suitable validity index needs to be selected, since there is no evaluation criterion that always leads to optimal results (Milligan and Cooper, 1985). In addition, the good performance of a validity criterion for certain data does not guarantee the same behaviour with different data (Xu and Wunsch II, 2005). And fourthly, if a distinct inflection point that suggests an optimal clustering solution has to be manually sought in the plotted results (as shown in Figure 2.4b), the final selection of the value of K will be actually performed by user instead of being solely driven by the data themselves, which may lead to non-optimal results biased by user's prior expectations (Everitt et al., 2011).

2.5.1.4 Self-refining consensus approach

As aforementioned in section 2.3.2, the proposal of the consensus clustering strategy consists of creating a cluster ensemble composed of a large number of individual clustering results and deriving a unique clustering solution upon that cluster ensemble through the application of a consensus clustering process. Obviously, the quality of the final clustering solution resulting from this consensus strategy can be negatively biased by the poor individual clustering solutions included in the cluster ensemble (Sevillano, 2009). In order to overcome this inconvenience, several consensus clustering solutions can be obtained by means of multiple consensus functions and a supraconsensus function can be applied with the aim of, in a blind manner, selecting the highest quality solution from the set of consensus clustering solutions previously generated (Strehl and Ghosh, 2002; Gionis et al., 2007).

In this context and closely related with the use of proper supraconsensus functions, the so-called consensus self-refining procedure is also proposed. Oriented to improve the quality of consensus clustering solutions, this self-refining strategy prove that it is likely to obtain, in a fully unsupervised fashion, a refined consensus clustering solution of equal (or even higher) quality than the best individual one (Fern and Lin, 2008) (Sevillano, 2009, Chapter 4, pages 111–132).

Thus, it seems plausible to define an approach based on a self-refining consensus procedure in order to provide a proper estimation of the real number of clusters in a dataset, since, similarly to the aforementioned relative validity approach, a correct choice of K would be subject to a correct selection of the best clustering solution in the cluster ensemble. However, not many consensus functions are capable of dealing with cluster ensemble components with distinct numbers of clusters, which, quite obviously, would be a unavoidable feature in such an approach (Sevillano, 2009).

In any case, this approach would present several drawbacks to deal with, such as the obvious rise of the computational cost of the process (beyond the cost of the consensus process, multiple clustering solutions are initially needed to build the cluster ensemble), the need of user intervention in the selection of the amount of components included in the cluster ensemble the self-refined consensus clustering is derived upon and the limited accuracy of the supraconsensus selection process, which is an area that requires of more research in order to be improved (Sevillano, 2009)

2.5.1.5 Model-based (or probabilistic) approach

The model-based approach to the problem of estimating the value of K is essentially based on the optimisation of some criterion functions under a probabilistic mixture-model framework. In a statistical framework, finding the correct number of clusters in a dataset is equivalent to fitting a model with observed data (different models correspond to clustering solutions with different values of K) and optimising some criterion (Xu and Wunsch II, 2005). Thus, the issue of the estimation of the number of clusters is transformed by the model-based clustering approach (see section 2.3.2) into a model selection problem in the probability framework (Gan et al., 2007).

On the one hand, the well-known EM algorithm (McLachlan and Krishnan, 1997) is habitually utilised to estimate the model parameters for a given K, which goes through a predefined range of values –the value of K that maximises (or minimises) the defined criterion function is regarded as optimal– (Xu and Wunsch II, 2005). Another well-known example implemented from this approach is the x-means algorithm (Pelleg and Moore, 2000), which is based on generating multiple clustering solutions with different values of K by the k-means algorithm and selecting the best one of them according to the criterion function.

On the other hand, a large number of criterion functions can be found in the literature, such as Akaike's information criterion (AIC) (Akaike, 1974; Windham and Culter, 1992), Bayesian information criterion (BIC) (Schwarz, 1978; Pelleg and Moore, 2000), minimum description length (MDL) (Rissanen, 1996; Grünwald et al., 1998), minimum message length (MML) (Oliver et al., 1996), cross validation-based information criterion (CVIC) (Smith and Hardaker, 2000b), or covariance inflation criterion (CIC) (Tibshirani and Knight, 1999), among others (McLachlan and Peel, 2000).

The main downsides associated to this approach lie on its computational requirements, since multiple clustering solutions are required in order to optimise the model; its high dependency on the selection of the model, which needs to fit the data –*i.e.* the model needs to be able of generating an optimal (or quasi-optimal) clustering solution for the correct value of K– (Fraley and Raftery, 1998); its dependency on the initialisation of the algorithm (EM or k-means, typically), which is closely related with the previous mentioned downside (Gan et al., 2007); and its dependency on a proper selection of the criterion function, being no criterion superior to others in general case –the selection of different criteria is still dependent on the data– (Xu and Wunsch II, 2005).

2.5.1.6 Adaptive approach

With the aim of overcoming some of the drawbacks of the previous approaches, it is worth considering a certain class of methods constituted by a diverse variety of clustering algorithms, which are implemented from a wide range of theoretical approaches to clustering (see section 2.3.2). The common characteristic of such clustering algorithms lies on the fact that they do not ask for the value of K as an input parameter. Instead of that, their behaviours are ruled over a set of configuration parameters, which is distinctive for each particular algorithm and which determines the final clustering solution the algorithm provides.

Hence, these so-called constructive clustering algorithms (Xu and Wunsch II, 2005) can adaptively and dynamically adjust the number of clusters of the clustering solution rather than use a previously specified and fixed value of K, which clearly defines a new different approach to the issue of determining the number of clusters in a dataset. Although such algorithms could be used in order to generate the diversity of clustering solutions required in aforementioned approaches by varying the values of their configuration parameters, their real goal is to have a sufficiently flexible behaviour to be able to provide by themselves optimal results in a diversity of clustering scenarios.

The only issue this class of clustering algorithms present lies on the fact that their configuration parameters need to be tuned by the user, which is a clear drawback, specially if heuristic knowledge (usually scenario-dependent) about the behaviour of the algorithm is not previously available. Thus, the problem of determining the number of clusters is converted by this approach into a parameter tuning problem, and the resulting number of clusters is largely dependent on parameter tweaking (Xu and Wunsch II, 2005).

Next, some of the most well-known examples of constructive clustering algorithms implemented from different theoretical approaches to clustering are listed, along with the detail of their distinctive configuration parameters:

Centre-based constructive clustering algorithms

- ISODATA algorithm (Ball and Hall, 1965) requires thresholds for the eleven parameters (maximum number of iterations, elongation criterion, closeness criterion, exclusion distance, minimum number of objects per cluster, etc.) that rule over its split-and-merge behaviour.
- HUFC algorithm (Geva, 1999) requires of the number of reduced features (F) after the PCA projection and the value of the parameter $(Constant)$ that weights the condition that determines if a cluster needs to be split (according to a fuzzy probabilistic approach based on a modified fuzzy k-means algorithm and the fuzzy maximum-likelihood estimation algorithm) in a lower level of the hierarchy (the outcome of the HUFC algorithm is a set of hierarchically structured fuzzy clusters).

Graph-based constructive clustering algorithms

- Chameleon algorithm (Karypis et al., 1999) requires the parameter (k) utilised to build the initial k-nearest neighbour graph, the minimum size $(MinSize)$ of the partitions obtained by splitting the k-nearest neighbour graph and the parameter (α) that weights the merging criterion the final clusters have to maximise.

Density-based constructive clustering algorithms

- DBSCAN algorithm (Ester et al., 1996) requires the radius (ϵ) utilised to estimate the density for each object in the dataset and the threshold $(MinPts)$ utilised to determine whether an object is a core point.

- DENCLUE algorithm (Hinneburg and Keim, 1998) requires the parameter (σ) that characterises the basic influence function and the threshold (ξ) utilised to decide whether an object is assigned to a certain density-attractor's cluster.

- VDBSCAN algorithm (Liu et al., 2007) improves DBSCAN by only requiring $MinPts$ and performing an intuitive calculation of several radius values (ϵ_i) that allow identifying clusters of different densities.

Grid-based constructive clustering algorithms

- OptiGrid algorithm (Hinneburg and Keim, 1999) requires the number of contracting projections (k) utilised to determine the cutting planes, the minimum cut score (mcs) utilised to select which is the best projection cut and the number of cutting planes (q) in the best projection cut utilised to determine the grids/clusters.

- WaveCluster (Sheikholeslami et al., 2000) algorithm requires the number of intervals each feature of the D-dimensional dataset is divided into $(m_i, \forall\, i \in \{1, D\})$ in the quantisation stage.

Combinatorial-based constructive clustering algorithms

- CLUSTERING algorithm (Tseng and Yang, 2001) requires the value of the parameter (w) that weights the fitness function that determines the resolution of the algorithm (a small value of w tends to increase both the number and the compactness of the clusters the algorithm identify).

Subspace constructive clustering algorithms

- CLIQUE algorithm (Agrawal et al., 1998) requires the number of intervals (ξ) each dimension of the dataset in partitioned into and the density threshold (τ) that determines whether a partition is dense enough or not.

- ENCLUS algorithm (Cheng et al., 1999) requires the entropy (ω) and the interest (ϵ) thresholds utilised to mine significant subspaces.

- FINDIT algorithm (Woo et al., 2004) requires the minimum size of clusters ($C_{minsize}$) and the minimum difference between two resultant clusters ($D_{mindist}$).

Kernel-based constructive clustering algorithms

- SVC algorithm (Ben-Hur et al., 2001) requires the width of the Gaussian kernel (q) that controls the scale at which the data is probed and the soft margin constant (C) that helps coping with outliers and overlapping clusters.

Neural networks-based constructive clustering algorithms

- ART algorithm (Carpenter and Grossberg, 1987) requires the confidence value (ρ) below which the match between the input pattern and the expectation has to be in order to generate a new cluster.

- SPLL algorithm (Zhang and Liu, 2002) requires the threshold value (ε) that rules over the splitting criterion for cluster prototypes.

2.5.1.7 Parameter-free approach

Most clustering algorithms implemented under the premises of the previous approaches require the setting of several input parameters and, therefore, the explicit intervention of user. As aforementioned, parameter-dependent clustering algorithms may fail in finding true clusters as a consequence of an incorrect parametrisation and lead to a clustering solution biased by user's subjective criteria. With the aim of being able to work without a prior specification of the true number of clusters in the dataset, as well as to not require any configuration parameter to be set by the user, different approaches to parameter-free clustering algorithms have been proposed over the last years. Thus, the main goal of parameter-free clustering algorithms consists in leading to a fully data-driven clustering solution, therefore avoiding user's ability to impose prejudices, expectations and presumptions on the clustering scenario.

A conceptual view of parameter-free DM applications of different nature (classification, clustering, anomalies detection, etc.) is provided by Keogh et al. (2004). They state that parameter-free approaches to DM present interesting advantages, such as their capability for allowing true exploratory data analysis (free of presumptions about the data) and their potentially superior accuracy, efficiency and generalisation ability in comparison to those of parameter-dependent approaches (even if exhaustive searches over parameters' values are performed). Their study is finally more focused on parameter-free optimal representations of the data than on parameter-free DM algorithms, so that they propose the Compression-Based Dissimilarity (CDM) measure, which is a parameter-free proximity measure based on information theory principles (specifically, on the Kolmogorov complexity theory) that can be applied on several DM scenarios.

First attempts of parameter-free clustering go back to the work of Gitman (1972), where a mathematical formulation of the clustering problem with no parameters controlled by user is presented. Such a proposal is actually a parameter-free relative validity approach to clustering, since it defines two different internal validity criteria from which determining an optimal partition –and therefore the value of K– out of a variety of clustering solutions: the maximisation of the average amount of structure (AS) and the maximisation of the average amount of stability (ST) –both of them combine measures of cohesion and isolation of the possible clusters in the dataset–.

Nonetheless, it has been recently when literature has provided a myriad of works focused on parameter-free clustering, which can be categorised into two main different approaches to the issue: **parameter-free split-and-merge approaches** and **parameter-free adaptive approaches**.

Parameter-free split-and-merge approaches

There exist a great amount of clustering methods that follow a split-and-merge strategy, which essentially consists on, firstly, splitting the data into a high number of small clusters and, secondly, merging some of these clusters according to some criterion (*e.g.* thresholding techniques, internal measures of clustering validity, optimisation cost functions) until a definitive clustering solution is finally obtained (Ding and He, 2002). Aside from parameter-dependent implementations of this strategy, such as the methods defined by Bajcsy and Ahuja (1998) or the improved version of the k-means algorithm proposed by Chen et al. (2004), the literature provides two different classes of parameter-free split-and-merge approaches to clustering:

1. **Parameter-free relative validity approach**. Usually from graph representations (such as dendrograms, minimum spanning trees, adjacency matrices, among others) of the relationships among the objects in the dataset, a relative validation (by means of the maximisation of some internal validity criterion, such as Silhouette Coefficient, quality indices for categorical data, or graph-based validity criteria, among others) of a set of clustering solutions is performed from this approach through a parameter-free split-and-merge procedure (Huang et al.,

2009; Raju and Kumari, 2011). Besides, some of the methods implemented under this approach are particularly designed in order to deal with some specific kinds of data, such as genomic data (Cesario et al., 2007; Bayá and Granitto, 2011) or protein-protein interactions (Ngomo, 2010).

2. **Parameter-free model-based approach**. A diversity of clustering solutions is evaluated in terms of the optimisation of a model-based criterion, usually designed from the information theory principle of the minimum description length (MDL), which determines that the best descriptor for a given set of data is the one that leads to the best compression of the data (Rissanen, 1996; Grünwald et al., 1998).

   A possible interpretation of this approach is made in terms of an improving of the results obtained by any given clustering algorithm. Once a clustering solution previously generated is taken, its clusters are firstly split into a higher amount of smaller clusters and, finally, some of these new clusters are merged into the final clusters of the solution. Böhm et al. (2007) propose a framework that, in addition, refines the split clusters with the aim of improving the performance of k-means, k-medoids and spectral clustering methods. Moreover, such a framework has been adapted to mixed attributed –numerical and categorical– datasets (Böhm et al., 2010).

   However, the great majority of algorithms implemented under this approach are designed from the graph-based cross-associations (CA) method proposed by Chakrabarti et al. (2004), which allows to decompose a binary matrix into disjoint row and column groups such that homogeneous intersections (clusters) are formed. Habitually, adjacency matrices (obtained from either a clustering algorithm or the proximity matrix of the data in the splitting stage) are used as graph representations of the data and a scalable parameter-free algorithm is run in the merging stage, so that the optimal CA –and therefore the optimal clustering solution obtained from the graph– is reached according to the optimisation of a MDL-based criterion (Chakrabarti, 2004). Many proposals in the literature follow this parameter-free strategy (Sun et al., 2007; He et al., 2009; Hirai et al., 2011; Mueller et al., 2011), or particularise it to specific kinds of data and/or applications such as image data organization for effective image retrieval (Oh et al., 2012). Furthermore, modified versions of this strategy have been also proposed both to find hierarchical clusters (Papadimitriou et al., 2008) and to perform hierarchical co-clustering –the final clustering solution results from crossing clusterings on both objects and features spaces– (Ienco et al., 2009). Finally, an evolved version of the method based on a greedy iterative heuristic procedure has been as well designed with the aim of find cohesive subgraphs (Akoglu et al., 2012).

Regardless of the specific kind of parameter-free split-and-merge approach each one of them belong to, all these clustering methods suffer from the same drawbacks; to wit, high dependance on both the clustering algorithm that generates the diversity of solutions and the validity indices utili-

sed to evaluate such solutions, the necessity of including the optimal –or a quasi-optimal– solution in the variety of evaluated clustering solutions and the high data-dependency in the selection of the model-based optimisation criterion.

Parameter-free adaptive approaches

The parameter-free adaptive approach is less habitual in the literature, while it is probably more interesting due to its ability to avoid the drawbacks presented by the parameter-free split-and-merge approach. The upside of the algorithms implemented from the parameter-free adaptive approach lies on the fact that they do not manipulate and/or evaluate representations of the data or clustering solutions previously generated by means of other procedures, but their own nature is free of tunable parameters and they are able to provide a final clustering solution without depending on any other methods. Mainly, three different classes of clustering methods have been so far proposed under this kind of approach:

1. **Parameter-free density-based approach**. While being the best-known density-based clustering method, DBSCAN algorithm (Ester et al., 1996) presents two distinct drawbacks: its dependency on two user-specified parameters and its inability to handle datasets with clusters with different densities. Hence, several methods have been proposed in order to address these problems, from among which the VDBSCAN algorithm (Liu et al., 2007) stands out, since it reduces the dependency to a single user-specified parameter and allows to identify clusters with different densities, as well as some derived methods that propose how to deal with the tuning of VDBSCAN's parameter, such as the subjective method designed by (Chowdhury et al., 2010) and the OVDBSCAN algorithm (Wang et al., 2013). Interesting surveys that study and compare the performances of these methods and some of the following have been conducted by Nagpal and Mann (2011) and Parimala et al. (2011).

   Along this line of work, parameter-free density-based algorithms have been implemented in order to overcome the problems presented by DBSCAN algorithm and its variants, as well as to eliminate any user intervention in the clustering process.

   Firstly, the DBCLASD algorithm (Xu et al., 1998) is a parameter-free density-based clustering algorithm proposed as an alternative to DBSCAN. DBCLASD combines its density-based nature with a model-based behaviour, since it defines the density of the clusters by estimating the probability distribution of the nearest neighbour distance within each cluster, which allows it to find clusters with different densities. Furthermore, a combination of the principles of DBCLASD and Chamaleon algorithms is proposed by Mu et al. (2008) in its parameter-free DMBC algorithm, although its experimental results has been tested from a very reduced set of datasets.

   And secondly, parameter-free improvements of the DBSCAN algorithm have been recently implemented by Chen et al. (2011) and Sabau (2012). Whereas the former generates a norma-

lised list of local densities to detect clusters with different densities in a purely density-based behaviour, the latter combines density-based clustering with a combinatorial search-based method with the aim of handling various data distributions, finding arbitrary shaped clusters and being robust to outliers.

In any case, all these methods (from the original DBSCAN algorithm and its variants, to the more recent parameter-free implementations) suffer from the same drawback, which seems to be inherent to their density-based nature: they fail to accurately delimit clusters whose objects are not uniformly distributed; *i.e.* they fail to properly identify clusters with variable density, since they characterise clusters according to their density and they interpret different densities as different clusters (Chen et al., 2011, Figure 11 and Table 3, page 985) (Sabau, 2012, Figure 8, page 205).

2. **Parameter-free AHC approach**. Another recent and very promising approach to parameter-free clustering is constituted by the algorithms proposed by Fred and Leitão (2003) and Aidos and Fred (2011a). By combining graph-based and model-based clustering concepts, they have implemented two different versions of a parameter-free AHC algorithm (SL-DID) according to merging decision criteria based on the distribution of the dissimilarity (or proximity) increments between neighbouring objects within a cluster –which has been also applied in a HPC version of the method (Aidos and Fred, 2011c)–. Although they obtain interesting results when clustering synthetic datasets with clusters of different nature, the performance of both algorithms worsens when dealing with real datasets.

   Since it is this precise approach which has been followed to implement the parameter-free AHC algorithm proposed in the present thesis (see Chapter 4 for further details), an insight into this issue is provided in Chapter 3, where both fundamentals on AHC methods and particulars of this parameter-free AHC approach are detailed.

3. **Miscellaneous methods**. Very recently, two new parameter-free clustering algorithms have been proposed from the adaptive strategy to the issue, but beyond the margins of both the density-based and the AHC approaches.

   Operating in a similar but completely inverse way to the split-and-merge method designed by Böhm et al. (2007), Xiong et al. (2012) propose the DHCC algorithm, a parameter-free DHC method for categorical data. According to a classical divisive strategy –*i.e.* starting from a single partition that includes all the objects in the dataset–, DHCC follows an iterative splitting procedure consisting of two phases per iteration: the bisection of a cluster is carried out in the preliminary splitting phase according to a MCA (multiple correspondence analysis for categorical data) criterion and the refinement phase optimises the global objective function SCE (sum of χ^2 error) locally to improve the quality of the bisection. The main issues of DHCC are its lack of generalisation, since it is particularised for categorical data, and its sensitivity to outlier objects (authors suggest the design of strategies to perform a previous detection of outliers as an interesting area for future research).

Cheung and Jia (2013) propose a penalised competitive learning clustering algorithm (PCL-OC) that, based on an object-cluster similarity metric for mixed data (both categorical and numerical attributed data), estimates the optimal value of K and provides a final HPC solution. PCL-OC requires two input parameters: a randomly initialised partition of K' $(K' > K)$ singleton clusters and the value of the learning rate (η) that regulates the cluster penalisation rule (redundant clusters are gradually faded out during the clustering process). Obviously, the result of the algorithm may depend on both the initial setting of K' and the objects random selection for the initial singleton clusters, specially in clustering scenarios with none previous knowledge about the data. Moreover, a heuristic-type method is provided to determine the correct value of η, but it is not clear how generalisable such a method can be.

2.6 Discussion

On the one hand, AHC methods seem to properly fit the problem of modelling learners' activity in online discussion forums when it is posed in clustering terms (see section 1.2.3.2), since the nature of such scenario allows both to take advantage of all AHC upsides –number of clusters not required in advanced, no initialisation needed and, above all, high suitability to exploratory data analysis– and, due to the relatively small size of the datasets such scenario involves, to avoid its main drawback –the computational cost– (see section 2.3.1 for further details).

On the other hand, as a first contribution of the present thesis, the different existing approaches that tackle the issue of the estimation of the optimal number of clusters in a dataset have been surveyed in section 2.5.1. Such a survey indicates that both the strategy habitually followed to model learners' activity in online discussion forums –*i.e.* relative validity approach– and the other strategies most typically adopted to handle this issue –*i.e.* heuristic-based approach, model-based approach, self-refining consensus approach, split-and-merge approaches, etc.– present noticeable drawbacks that, depending on the characteristics of the clustering scenario, can easily make them unable to provide optimal clustering results. Therefore, all these approaches certainly leave room for improvement regarding this particular task.

Consequently, a survey specifically concerning AHC methods is performed in the next chapter of the present thesis, whose main focuses are the fundamentals of AHC algorithms, their appropriateness for determining the number of clusters in a dataset and the particularities, advantages and drawbacks of the parameter-free AHC algorithms proposed in the literature to date (see section 2.5.1.7).

Chapter 3

Overview of AHC algorithms

Given their strong points (see section 2.3.1), AHC methods are a natural choice in the clustering scenario derived from modelling learners' activity in online discussion forums (see section 1.2.3.2). Moreover, the diversity of approaches to the issue of determining the number of clusters in a dataset (see section 2.5.1) reveals the usage of parameter-free AHC algorithms as an interesting and promising strategy in this context, since they combine their behaviour free of tunable parameters (therefore avoiding any user intervention in the clustering stage) with the rest of AHC upsides. Hence, the present thesis requires a more detailed survey on the nature of AHC and more particularly on the specifics of parameter-free AHC algorithms, in order to clearly identify all their benefits and to understand how to improve their limitations, if necessary. Thus, the goals of the present chapter are to deepen the fundamental principles of AHC methods, to determine their usefulness in the estimation of the optimal number of clusters and to study whether existing approaches to parameter-free AHC are able to deal with the particularities of the online forums activity modelling scenario or not.

Consequently, this third chapter is firstly focused on the fundamentals of AHC methods, which are surveyed in section 3.1, laying special emphasis on how to postprocess dendrograms to try to determine the real number of clusters in a dataset, as well as to obtain a derived HPC solution. Next, the particularities of parameter-free approaches to AHC are tackled in section 3.2. Finally, the upsides and limitations of the currently existing parameter-free AHC algorithms regarding their possible application to the analysis of learners' activity in online discussion forums are discussed in section 3.3.

3.1 Fundamentals on AHC methods

The basics of the AHC methods, the characteristics of the main AHC algorithms and the procedures to obtain a HPC solution from a dendrogram are studied in the present section. Thus, although both terms are habitually used with the same purpose, it is worth noting the differences between method, which involves a general description and a target structure, and algorithm, which is an specific implementation of a method in terms of both efficiency –from the computational point of view– and effectiveness –from the application point of view– (Murtagh and Contreras, 2012).

Next, the most common and most widely used scheme of the AHC methods is introduced. Nonetheless, there exist other strategies that allow building a hierarchy of clusters. Sneath and Sokal (1973) may be also consulted for other representations of the stages in the construction of HC methods. And, in general terms, a further insight into AHC and other HC methods can be gained by referring to Gordon (1987), Holman (1992), Gordon (1996), Gan et al. (2007), Carlsson and Mémoli (2009) and Murtagh and Contreras (2012).

Thus, let $\mathbf{X}$ be a dataset constituted by a set of N D-dimensional objects. As it has been previously illustrated in Figure 2.1, AHC methods start with every single object in a single cluster (N singleton clusters) and perform a series of merging operations ($N-1$ merging steps) so that the closest pair of clusters according to some cluster proximity criterion (*i.e.* linkage function) are merged at every step (leading to $N-1$ overlapped clusters) until all the objects are held under one cluster (Gan et al., 2007). Thus, according to equation 2.15, an AHC solution consists of a set $\mathbf{P}$ of $2N-1$ overlapped clusters represented by means of a dendrogram $\mathbf{\Delta}$ (see Figure 2.2b). Hence, the general AHC procedure can be summarised by the method shown in Algorithm 2.

Algorithm 2 Basic AHC method (adapted from Xu and Wunsch II (2005) and Tan et al. (2006)).

1: *Input*: Dataset $\mathbf{X}$
2: **procedure**
3: $M_P : M_P(i,j) \leftarrow d_{\mathbf{x}_i\mathbf{x}_j}, \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X} \mid i \neq j$ ▷ Initially, $d_{ij} = d_{\mathbf{x}_i\mathbf{x}_j}$, since $\mathbf{P} = \{\mathbf{C}_1, ..., \mathbf{C}_N\} \mid \mathbf{C}_i = \{\mathbf{x}_i\}$
4: **for** $n \leftarrow 1, N-1$ **do**
5: Select the two closest clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ according to M_P ▷ $d_{N+n} = d_{ij}$
6: $\mathbf{\Delta} : \overline{\Delta}_n \leftarrow [i\ j\ d_{N+n}]$ ▷ $\mathbf{P} = \mathbf{P} + \mathbf{C}_{N+n} \mid \mathbf{C}_{N+n} = \mathbf{C}_i \cup \mathbf{C}_j$
7: Update M_P: According to some linkage function, reflect the proximity between the new cluster $\mathbf{C}_{N+n}$ and the rest of clusters, and dismiss the proximities between $\mathbf{C}_i$, $\mathbf{C}_j$ and the rest of clusters.
8: **end for**
9: **end procedure**
10: *Output*: Dendrogram $\mathbf{\Delta}$ ▷ $\mathbf{\Delta}$ represents the final set $\mathbf{P}$ of $2N-1$ clusters

As shown in line 3, the first step defined in the method involves calculating the proximity matrix (M_P) of $\mathbf{X}$. As aforementioned in section 2.2 concerning equation 2.12, M_P contains $\frac{N^2-N}{2}$ different proximity values, so that the minimum computational requirements of this method grow quadratically with N, which is the reason why AHC methods are, at least, $O(N^2)$ –if a sorting of the values in M_P prior to the execution of the procedure was required, the method would then be $O(N^2 \log N)$–.

Moreover, once two clusters have merged giving rise to a new cluster in the hierarchy, the proximities between the new cluster and the rest of clusters in $\mathbf{P}$ need to be computed (see line 7 in Algorithm 2). Hence, a linkage function that defines how to update M_P needs to be defined (Gan et al., 2007). To that effect, Lance and Williams (1967) proposed the Lance-Williams recurrence update formula, which gives the proximity between a cluster $\mathbf{C}_l$ and a cluster $\mathbf{C}_k$ resultant of the union of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$:

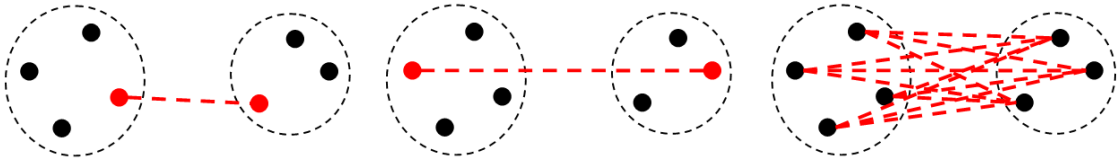
$$d_{kl} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \alpha_i d_{il} + \alpha_j d_{jl} + \beta d_{ij} + \gamma \left| d_{il} - d_{jl} \right| \tag{3.1}$$

Different configurations of the parameters α_i, α_j, β and γ in the Lance-Williams formula define different linkage functions, which in its turn give rise to different variants of the basic AHC method (Murtagh and Contreras, 2012, Table 1, page 6). Some properties of Lance-Williams formula have been investigated by DuBien and Warde (1979) and a more general recurrence formula with more configuration parameters has been proposed by Jambu (1978). In addition, there exist AHC methods less used in practise that do not fit the Lance-Williams formula, such as the bicriterion analysis proposed by Delattre and Hansen (1980).

Therefore, regarding the way their linkage functions are defined, basic AHC methods can be conveniently classified into two main groups: the first group is that whose linkage functions are defined from graph-based criteria (see section 3.1.1) and the second group is that whose linkage functions are based on geometric concepts which allow the cluster centres to be specified (see section 3.1.2). Other methods different from those detailed in the two following sections have been defined by Scheibler and Schneider (1985) and Scheibler and Schneider (1989) from the more general recurrence formula proposed by Jambu (1978). Finally, different strategies in order to postprocess the dendrogram and obtain an HPC solution are presented in section 3.1.3.

3.1.1 Graph-based AHC methods

In graph-based AHC methods, the proximity between two clusters is defined by means of linkage criteria that represent each cluster in terms of a subgraph or interconnected points Gan et al. (2007). The three most common graph-based linkage criteria according to literature are illustrated in Figure 3.1, which give rise to the three more widely used graph-based AHC methods: Single-Link method (see section 3.1.1.1), Complete-Link method (see section 3.1.1.2) and Average-Link method (see section 3.1.1.3).



(a) Single-Link criterion. ***(b)*** Complete-Link criterion. ***(c)*** Average-Link criterion.

Figure 3.1: Graph-based linkage criteria.

3.1.1.1 Single-Link (or nearest neighbour) method

First introduced by Florek et al. (1951) and, later, independently by McQuitty (1957) and Sneath (1957), the Single-Link (SL) method is the simplest and most widely used AHC method. It establishes that the proximity between to exclusive clusters $\mathbf{C}_k$ and $\mathbf{C}_l$ equals to the proximity between the two closest objects $\mathbf{x}_m$ and $\mathbf{x}_p$ respectively belonging to $\mathbf{C}_k$ and $\mathbf{C}_l$; *i.e.* the proximity between clusters is determined by the nearest neighbour criterion (Gan et al., 2007):

$$d_{kl}^{(SL)} = d_{\mathbf{x}_m\mathbf{x}_p} = \min\left\{d_{\mathbf{x}_q\mathbf{x}_r}\right\}, \forall\, \mathbf{x}_q \in \mathbf{C}_k, \forall\, \mathbf{x}_r \in \mathbf{C}_l, \forall\, \mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.2}$$

The linkage function defined by the SL method can be derived by particularising the parameters in Lance-Williams formula according to the values $\alpha_i = \frac{1}{2}$, $\alpha_j = \frac{1}{2}$, $\beta = 0$ and $\gamma = -\frac{1}{2}$ (Murtagh and Contreras, 2012):

$$d_{kl}^{(SL)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{1}{2}d_{il} + \frac{1}{2}d_{jl} - \frac{1}{2}\left|d_{il} - d_{jl}\right| = \min\left\{d_{il}, d_{jl}\right\} \tag{3.3}$$

SL method gives priority to the connectedness between objects, so that it allows to identify unbalanced clusters with arbitrary shapes and distributions, although it tends to present a low robustness against noise (Everitt et al., 2011). The SLINK algorithm proposed by Sibson (1973) is a $O(N^2)$ implementation of the SL method. Furthermore, other $O(N^2)$ time and $O(N^2)$ space implementations are described in Murtagh (1983) and Murtagh (1985).

3.1.1.2 Complete-Link (or farthest neighbour) method

First introduced by Sørensen (1948) and, later, independently by Johnson (1967) and Lance and Williams (1967), the Complete-Link (CL) method establishes that the proximity between to exclusive clusters $\mathbf{C}_k$ and $\mathbf{C}_l$ equals to the proximity between the two most distant objects $\mathbf{x}_m$ and $\mathbf{x}_p$ respectively belonging to $\mathbf{C}_k$ and $\mathbf{C}_l$; *i.e.* the proximity between clusters is determined by the farthest neighbour criterion (Gan et al., 2007):

$$d_{kl}^{(CL)} = d_{\mathbf{x}_m\mathbf{x}_p} = \max\left\{d_{\mathbf{x}_q\mathbf{x}_r}\right\}, \forall\, \mathbf{x}_q \in \mathbf{C}_k, \forall\, \mathbf{x}_r \in \mathbf{C}_l, \forall\, \mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.4}$$

The linkage function defined by the CL method can be derived by particularising the parameters in Lance-Williams formula according to the values $\alpha_i = \frac{1}{2}$, $\alpha_j = \frac{1}{2}$, $\beta = 0$ and $\gamma = \frac{1}{2}$ (Murtagh and

Contreras, 2012):

$$d_{kl}^{(CL)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{1}{2}d_{il} + \frac{1}{2}d_{jl} + \frac{1}{2}\left|d_{il} - d_{jl}\right| = \max\left\{d_{il}, d_{jl}\right\} \tag{3.5}$$

CL method gives priority to the compactness of the sets of objects, so that it tends to identify compact balanced clusters with equal diameters. It also tends to be more robust against noise than the SL method (Everitt et al., 2011). The CLINK algorithm proposed by Defays (1977) is a $O\left(N^2\right)$ implementation of the CL method. Furthermore, other $O\left(N^2\right)$ time and $O\left(N^2\right)$ space implementations are described in Murtagh (1983) and Murtagh (1985).

3.1.1.3 Average-Link (or group average) method

First introduced by Sokal and Michener (1958) and, later, independently by Lance and Williams (1966) and McQuitty (1966), the Average-Link (AL) method establishes that the proximity between to exclusive clusters $\mathbf{C}_k$ and $\mathbf{C}_l$ equals to the average of the proximities between all possible pairs of objects that are made up of one object from each cluster; *i.e.* the proximity between clusters is determined by the group average criterion (Gan et al., 2007). There exist two different variants of the AL criterion; the UPGMA method (unweighted pair group method using arithmetic averages):

$$d_{kl}^{(UPGMA)} = \frac{1}{N_k N_l}\sum_{q=n_1^k}^{n_{N_k}^k}\sum_{r=n_1^l}^{n_{N_l}^l} d_{\mathbf{x}_q\mathbf{x}_r},\ \forall\,\mathbf{x}_q \in \mathbf{C}_k,\ \forall\,\mathbf{x}_r \in \mathbf{C}_l,\ \forall\,\mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.6}$$

where n_m^p the index of the mth object belonging to cluster $\mathbf{C}_p$, and N_k and N_l are the number of objects in clusters $\mathbf{C}_k$ and $\mathbf{C}_k$, respectively; and the WPGMA method (weighted pair group method using arithmetic averages):

$$d_{kl}^{(WPGMA)} = \sum_{q=n_1^k}^{n_{N_k}^k}\sum_{r=n_1^l}^{n_{N_l}^l}\left(\frac{1}{2}\right)^{\alpha_q^k+\alpha_r^l} d_{\mathbf{x}_q\mathbf{x}_r},\ \forall\,\mathbf{x}_q \in \mathbf{C}_k,\ \forall\,\mathbf{x}_r \in \mathbf{C}_l,\ \forall\,\mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.7}$$

where α_m^p is the number of subclusters nested under the cluster $\mathbf{C}_p$ (including singleton clusters) the object $\mathbf{x}_m$ belongs to.

The linkage function defined by the UPGMA method can be derived by particularising the parameters in Lance-Williams formula according to the values $\alpha_i = \frac{N_i}{N_k}$, $\alpha_j = \frac{N_j}{N_k}$, $\beta = 0$ and $\gamma = 0$ (Murtagh and Contreras, 2012):

$$d_{kl}^{(UPGMA)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{N_i}{N_k}d_{il} + \frac{N_j}{N_k}d_{jl} \tag{3.8}$$

whereas, in the case of the WPGMA method, the values of the parameters are $\alpha_i = \frac{1}{2}, \alpha_j = \frac{1}{2}, \beta = 0$ and $\gamma = 0$:

$$d_{kl}^{(WPGMA)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{1}{2}d_{il} + \frac{1}{2}d_{jl} \tag{3.9}$$

AL methods tend to join clusters with small variances and, although they also tend to give more priority to the compactness than to the connectedness, they are usually considered as an intermediate option between SL and CL methods. Whereas WPGMA tends to identify more unbalanced

clusters, since it weights objects in small clusters more highly than UPGMA, both methods are relatively robust against noise (Everitt et al., 2011). Murtagh (1983) and Murtagh (1985) describe $O(N^2)$ time and $O(N^2)$ space implementations of both UPGMA and WPGMA methods.

3.1.2 Geometric AHC methods

In geometric AHC methods, the proximity between two clusters is defined by means of linkage criteria that represent each cluster in terms of its centroid –or centre point– Gan et al. (2007). The three most widely used geometric AHC methods according to literature are Centroid method (see section 3.1.2.1), Median method (see section 3.1.2.2) and Ward's method (see section 3.1.2.3). Although all these methods use the proximity between clusters' centroids in order to define the proximity between clusters (see Figure 3.2), they present significant differences relating to both the way the centroid of a new cluster is located and the way the proximity between two clusters is measured from their centroids.

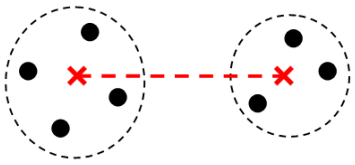


Figure 3.2: Geometric linkage criteria.

It is worth noting that geometric AHC methods are less used in practise that graph-based AHC methods, since, among other reasons, they only support proximity measures based on representation spaces geometrically interpretable –*e.g.* Euclidean space–. Moreover, Centroid and Median methods may give rise, unlike the rest of AHC methods detailed in the present thesis, to non-monotonic dendrograms (Everitt et al., 2011); *i.e.* Centroid and Median methods do not guarantee that the proximity level of any given cluster in the hierarchy is higher than those of its nested subclusters ($n_i > n_j \nLeftrightarrow d_{N+n_i} \geq d_{N+n_j}$; see Figure 3.3k for further details). Finally, discussions on $O(N^2)$ time and $O(N)$ space implementations of the geometric AHC methods next detailed can be found in Murtagh (1983), Day and Edelsbrunner (1984), Murtagh (1984) and Murtagh (1985).

3.1.2.1 Centroid (or UPGMC) method

First introduced by Sokal and Michener (1958), the Centroid method (or UPGMC –unweighted pair group method using centroids–) establishes that the proximity between to exclusive clusters $\mathbf{C}_k$ and $\mathbf{C}_l$ equals to the proximity between their two respective centroids (Gan et al., 2007):

$$d_{kl}^{(UPGMC)} = d_{\mathbf{g}_k \mathbf{g}_l}, \forall \mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.10}$$

where $\mathbf{g}_k$ and $\mathbf{g}_l$ are the centroids of clusters $\mathbf{C}_k$ and $\mathbf{C}_l$, respectively.

The linkage function defined by the UPGMC method can be derived by particularising the parameters in Lance-Williams formula according to the values $\alpha_i = \frac{N_i}{N_k}$, $\alpha_j = \frac{N_j}{N_k}$, $\beta = -\frac{N_i N_j}{N_k^2}$ and $\gamma = 0$ (Murtagh and Contreras, 2012):

$$d_{kl}^{(UPGMC)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{N_i}{N_k} d_{il} + \frac{N_j}{N_k} d_{jl} - \frac{N_i N_j}{N_k^2} d_{ij} \tag{3.11}$$

where N_i, N_j and N_k are the number of objects in clusters $\mathbf{C}_i$, $\mathbf{C}_j$ and $\mathbf{C}_k$, respectively.

In UPGMC method, the centroid of the merged cluster $(\mathbf{g}_k)$ tends to be dominated by the centroid of the more numerous subcluster, since $\mathbf{g}_k = \frac{N_i \mathbf{g}_i + N_j \mathbf{g}_j}{N_k}$, where $\mathbf{g}_i$ and $\mathbf{g}_j$ are the centroids of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively.

3.1.2.2 Median (or WPGMC) method

First introduced by Gower (1967), the Median method (or WPGMC –weighted pair group method using centroids–) establishes that the proximity between to exclusive clusters $\mathbf{C}_k$ and $\mathbf{C}_l$ equals to the proximity between their two respective centroids (Gan et al., 2007):

$$d_{kl}^{(WPGMC)} = d_{\mathbf{g}_k \mathbf{g}_l}, \forall\, \mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.12}$$

where $\mathbf{g}_k$ and $\mathbf{g}_l$ are the centroids of clusters $\mathbf{C}_k$ and $\mathbf{C}_l$, respectively.

The linkage function defined by the WPGMC method can be derived by particularising the parameters in Lance-Williams formula according to the values $\alpha_i = \frac{1}{2}$, $\alpha_j = \frac{1}{2}$, $\beta = -\frac{1}{4}$ and $\gamma = 0$ (Murtagh and Contreras, 2012):

$$d_{kl}^{(WPGMC)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{1}{2} d_{il} + \frac{1}{2} d_{jl} - \frac{1}{4} d_{ij} \tag{3.13}$$

In WPGMC method, the centroid of the merged cluster $(\mathbf{g}_k)$ is in an intermediate position between the centroids of both subclusters, since $\mathbf{g}_k = \frac{\mathbf{g}_i + \mathbf{g}_j}{2}$, where $\mathbf{g}_i$ and $\mathbf{g}_j$ are the centroids of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively.

3.1.2.3 Ward's (or minimum variance) method

Proposed by Ward Jr. (1963) and Ward Jr. and Hook (1963), Ward's method minimises the loss of information associated with the merging of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ into the new cluster $\mathbf{C}_k$. Ward's method quantifies the information loss by means of an error sum of squares (ESS) local objective function, so that it is often referred to as the minimum variance method (Gan et al., 2007):

$$ESS\left(\mathbf{C}_i\right) = \sum_{\mathbf{x} \in \mathbf{C}_i} \left(\mathbf{x} - \mathbf{g}_i\right) \left(\mathbf{x} - \mathbf{g}_i\right)^T \tag{3.14}$$

where ^{T} denotes transposition, $\mathbf{g}_i$ is the centroid of $\mathbf{C}_i$ and $ESS\left(\mathbf{C}_i\right)$ is the ESS of $\mathbf{C}_i$. Thus, the total within-cluster ESS is the objective function minimised at the nth step of the method:

$$ESS = \sum_{i=N+1}^{N+n} ESS\left(\mathbf{C}_i\right) \tag{3.15}$$

Ward's method establishes that the proximity between to exclusive clusters $\mathbf{C}_k$ and $\mathbf{C}_l$ equals to the weighted proximity between their two respective centroids (Gan et al., 2007):

$$d_{kl}^{(Ward's)} = \frac{N_k N_l}{N_k + N_l} d_{\mathbf{g}_k \mathbf{g}_l}, \forall\, \mathbf{C}_k \cap \mathbf{C}_l = \emptyset \tag{3.16}$$

where N_k and N_l, on the one hand, and $\mathbf{g}_k$ and $\mathbf{g}_l$, on the other hand, are both the number of objects and the centroids of clusters $\mathbf{C}_k$ and $\mathbf{C}_l$, respectively.

The linkage function defined by the Ward's method can be derived by particularising the parameters in Lance-Williams formula according to the values $\alpha_i = \frac{N_i+N_l}{N_k+N_l}$, $\alpha_j = \frac{N_j+N_l}{N_k+N_l}$, $\beta = -\frac{N_l}{(N_k+N_l)^2}$ and $\gamma = 0$ (Murtagh and Contreras, 2012):

$$d_{kl}^{(Ward's)} = D\left(\mathbf{C}_i \cup \mathbf{C}_j, \mathbf{C}_l\right) = \frac{N_i + N_l}{N_k + N_l} d_{il} + \frac{N_j + N_l}{N_k + N_l} d_{jl} - \frac{N_l}{\left(N_k + N_l\right)^2} d_{ij} \tag{3.17}$$

where N_i, N_j, N_k, and N_l are the number of objects in clusters $\mathbf{C}_i$, $\mathbf{C}_j$, $\mathbf{C}_k$ and $\mathbf{C}_l$, respectively.

In Ward's method, the centroid of the merged cluster $(\mathbf{g}_k)$ is located as in UPGMC method, *i.e.* $\mathbf{g}_k = \frac{N_i \mathbf{g}_i + N_j \mathbf{g}_j}{N_k}$, where $\mathbf{g}_i$ and $\mathbf{g}_j$ are the centroids of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively. Furthermore, Ward's method tends to identify balanced and spherical clusters.

3.1.3 Obtaining data partitions from a dendrogram

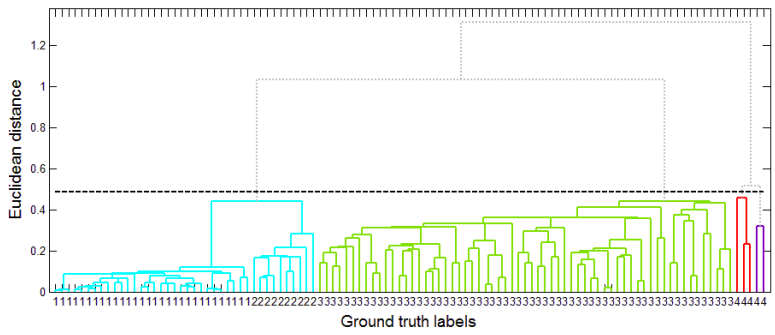
As aforementioned in section 2.3, the clustering solution resulting from an AHC algorithm is usually depicted by a dendrogram, which is a representation that provides very informative descriptions and visualisation for the potential data clustering structures (Xu and Wunsch II, 2005). Nonetheless, since a HC solution is nothing more than a sequence of hierarchical partitions of the data, there exist different strategies that allow postprocessing the dendrogram in order to obtain an HPC solution (*i.e.* obtaining a cluster label vector $\boldsymbol{\lambda}$ from a dendrogram $\boldsymbol{\Delta}$).

The first and most common approach to this issue is to derive a partition by simply cutting the dendrogram at a certain proximity level (Murtagh and Contreras, 2012); *i.e.* to dismiss those clusters (links in the dendrogram) whose proximity level is higher than some threshold $(d_i > d_{th})$. Thus, setting d_{th} in order to perform the cut is a parameter selection problem, which can be posed either previously determining the number of clusters in the HPC solution $(K \rightarrow d_{th} \rightarrow \boldsymbol{\lambda})$ or deriving some criterion to determine d_{th} from the information on the data provided by the dendrogram $(\boldsymbol{\Delta} \rightarrow d_{th} \rightarrow \boldsymbol{\lambda})$. Whichever strategy applies, this approach always presents the same problem as to the obtainment of an optimal HPC solution: the dendrogram has to contain the real clusters

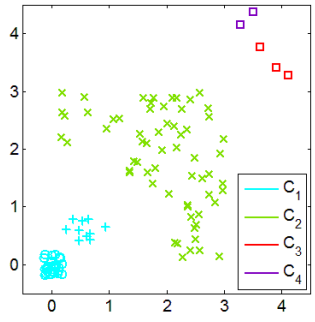
and they have to be the K clusters present at the Kth level in the hierarchy (*i.e.* the $(N-K)$th step in Algorithm 2), otherwise they cannot be identified by a horizontal cut in the dendrogram.

Figure 3.3 illustrates the particularities and limitations of this approach by applying the seven different AHC methods detailed in sections 3.1.1 and 3.1.2 to the *4toy* dataset presented in Figure 2.3a. Since this dataset comprises 4 real clusters, every dendrogram has been cut at a d_{th} calculated so that a 4-cluster HPC solution is obtained. Different considerations can be stated:

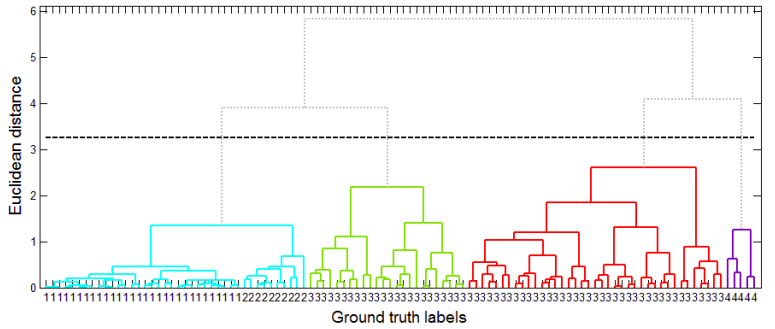
- The limitations of this approach are clearly revealed by the fact that none of the seven methods leads to a dendrogram that allows recovering the 4 real clusters by means of any horizontal cut and therefore identifying the optimal HPC solution.
- It is also worth noting that SL, UPGMA, WPGMA and UPGMC methods lead to dendrograms that, while at different levels in the hierarchy, do contain the 4 real clusters present in the



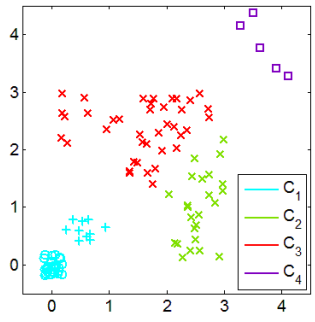
(a) SL: Dendrogram ($CPCC = 0.8$; $d_{th} = 0.49$).



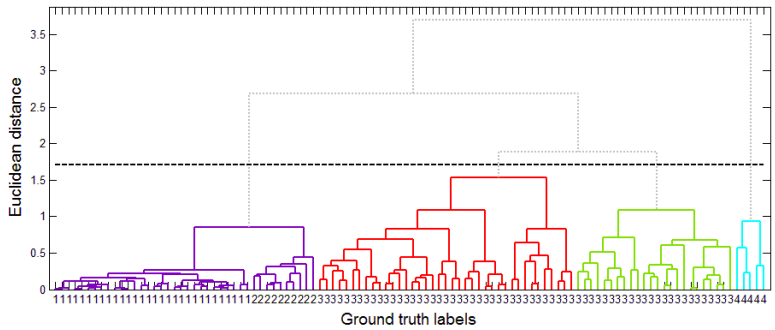
(b) SL: HPC solution.



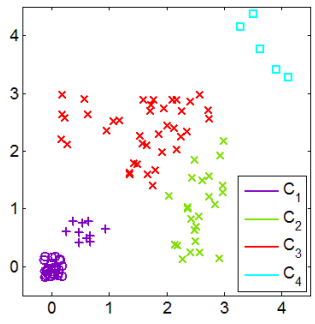
(c) CL: Dendrogram ($CPCC = 0.72$; $d_{th} = 3.3$).



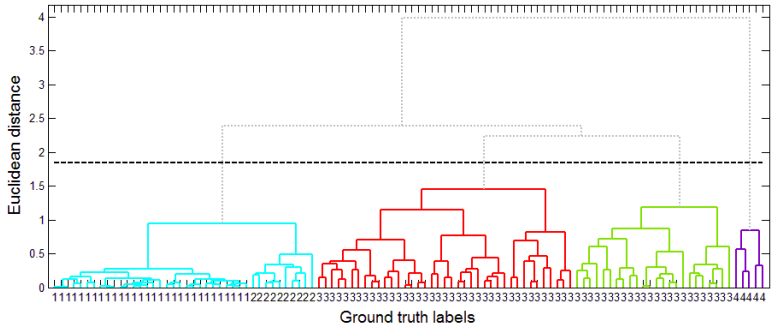
(d) CL: HPC solution.



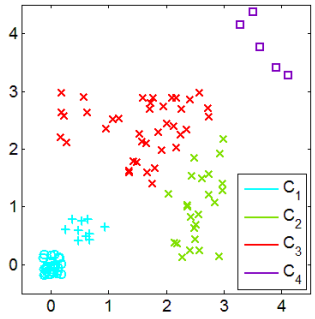
(e) UPGMA: Dendrogram ($CPCC = 0.85$; $d_{th} = 1.7$).



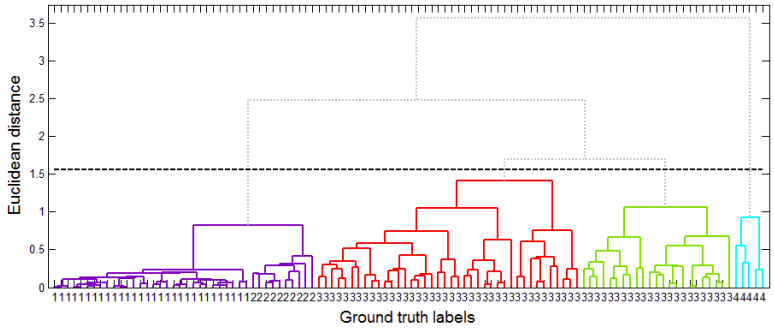
(f) UPGMA: HPC solution.



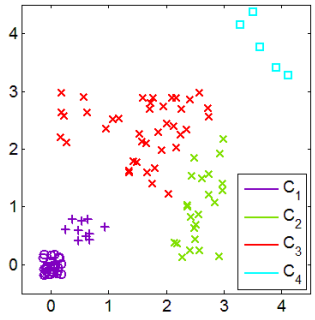
(g) WPGMA: Dendrogram ($CPCC = 0.82$; $d_{th} = 1.8$).



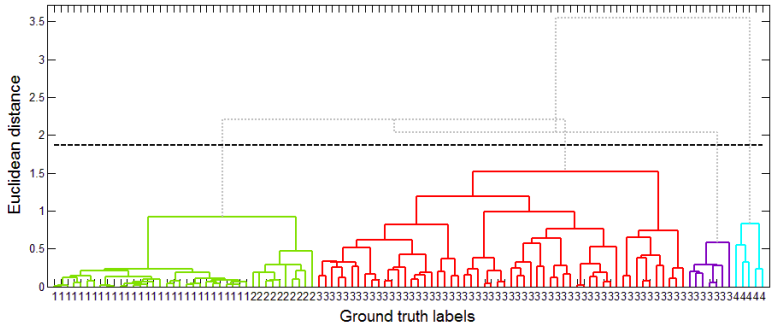
(h) WPGMA: HPC solution.



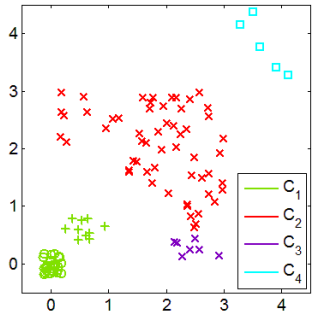
(i) UPGMC: Dendrogram ($CPCC = 0.85$; $d_{th} = 1.6$).



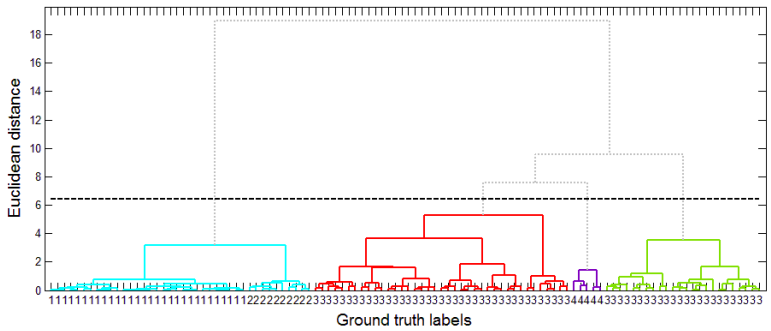
(j) UPGMC: HPC solution.



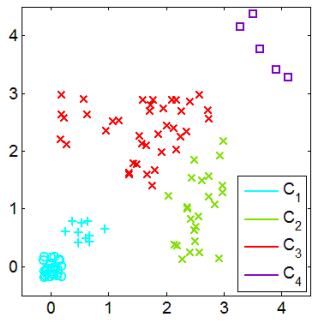
(k) WPGMC: Dendrogram ($CPCC = 0.83$; $d_{th} = 1.9$).



(l) WPGMC: HPC solution.



(m) Ward's: Dendrogram ($CPCC = 0.77$; $d_{th} = 6.04$).



(n) Ward's: HPC solution.

Figure 3.3: HC and HPC solutions on the *4toy* dataset by means of graph-based AHC methods. ***(a) (c) (e) (g) (i) (k) (m)*** Clusters over the horizontal cut in the dendrogram (black dashed line) are dismissed (grey dotted links) to obtain a 4-cluster HPC solution. Ground truth's cluster labels are shown on the x-axis, as well as Cophenetic Correlation Coefficients ($CPCC$) and the proximity threshold levels (d_{th}). ***(b) (d) (f) (h) (j) (l) (n)*** Scatterplots: clusters of the HPC solution are depicted by colours (see legends).

dataset, which makes these dendrograms better HC solutions than the rest, since they may allow other more sophisticated approaches to identify the optimal HPC solution.

- Finally, the $CPCC$ of the different dendrograms have been calculated before the cutting process. As aforementioned in section 2.4.4, the obtained range of $CPCC$ values ($[0.72, 0.85]$) indicates that $CPCC$ does not adopt significantly high or low values depending on whether the dendrogram contains the 4 real clusters or not, respectively. Nonetheless, it can be interpreted a slight tendency to lower $CPCC$ values in the dendrograms that do not even contain the 4 real clusters at different levels in the hierarchy (CL and Ward's present the lowest $CPCC$ values, while WPGMC dendrogram breaks this tendency).

Hence, considering the shortcomings of this strategy, other approaches arise aiming to detect clusters of interest at varying levels of the hierarchy (Murtagh and Contreras, 2012). One of the most widely used is Zahn's inconsistency criterion (ZIC), which establishes that, instead of determining an horizontal cutting threshold, the dendrogram should be cut at its most inconsistent links (or edges, in a minimum spanning tree), so that the most inconsistent clusters are dismissed, as well as their superior clusters in the hierarchy (Zahn, 1971).

Given a cluster $\mathbf{C}_i$ and a sample $\overline{d}_i$ consisting on its own proximity level (d_i) and the proximity levels of its most immediate nested clusters in the hierarchy $\left(\overline{d}_{N+i} = [d_{N+i}\, d_{\Delta_{i1}}\, d_{\Delta_{i2}} ...]\right)$, ZIC defines the inconsistency of cluster $\mathbf{C}_i$ (ι_i) as the extent d_i exceeds the sample average of $\overline{d}_i$ measured in units of the sample standard deviation of $\overline{d}_i$; *i.e.* ι_i is defined as the standard score of d_i with respect to $\overline{d}_i$ (Zahn, 1971):

$$\iota_i = \frac{d_i - \mu_{\overline{d}_i}}{\sigma_{\overline{d}_i}} \tag{3.18}$$

being $\mu_{\overline{d}_i}$ and $\sigma_{\overline{d}_i}$ the sample average and the sample standard deviation of $\overline{d}_i$, respectively:

$$\mu_{\overline{d}_i} = \frac{1}{M}\sum_{j=1}^{M} \overline{d}_{i_j} \quad , \quad \sigma_{\overline{d}_i} = \frac{1}{M}\sum_{j=1}^{M} \left(\overline{d}_{i_j} - \mu_{\overline{d}_i}\right)^2 \tag{3.19}$$

where $\overline{d}_{i_j}$ and M are the jth observation and the length of sample $\overline{d}_i$, respectively. Since observations in sample $\overline{d}_i$ are taken from a subtree of depth δ (a user-selected parameter that indicates how many hierarchical levels of nested clusters under $\mathbf{C}_i$ are considered to obtain $\overline{d}_i$), the length of sample $\overline{d}_i$ directly depends on δ $\left(M = 2^\delta - 1\right)$.

Thus, once the dendrogram is completed according to Algorithm 2 and the inconsistency of its links is calculated according to equation 3.18, a HPC solution can be obtained by cutting the dendrogram at a certain inconsistency level; *i.e.* by dismiss those links whose inconsistency value is higher than some threshold $(\iota_i > \iota_{th})$ and their higher (in proximity terms) links. Again, parameter ι_{th} can be determined either from a target value of K $(K \rightarrow \iota_{th} \rightarrow \boldsymbol{\lambda})$ or from the information provided by the dendrogram $(\boldsymbol{\Delta} \rightarrow \iota_{th} \rightarrow \boldsymbol{\lambda})$.

Figure 3.4 illustrates how ZIC allows to obtain optimal HPC solutions from the four AHC methods (SL, UPGMA, WPGMA and UPGMC) whose dendrograms (see Figure 3.3) contain the 4 real clusters

present in the *4toy* dataset at different levels in the hierarchy. Some conclusions about ZIC can be obtained regarding these results:

- Although this approach does not need the dendrogram to include the optimal solution at some level in the hierarchy, it does need the K optimal clusters to be preserved under the most inconsistent links in the dendrogram. Hence, its performance still depends on obtaining a well-aimed dendrogram in the first place.
- Since ZIC performs a local measure for the cluster inconsistency, the values of ι_i and therefore the compliance of the condition mentioned in the previous point depend in great extent on a proper selection of δ. Hence, while it can lead to better clustering results, this approach involves a more complex parameter selection problem, given that two parameters are now required to be successfully estimated: the proximity levels sample's depth (δ) and the inconsistency threshold (ι_{th}).
- Finally, having obtained an optimal HPC solution by means of four different AHC methods, each one of the 4 clusters in the dataset is represented by four different subdendrograms. As shown in Table 3.1, a $CPCC$ can be calculated for each one of them.

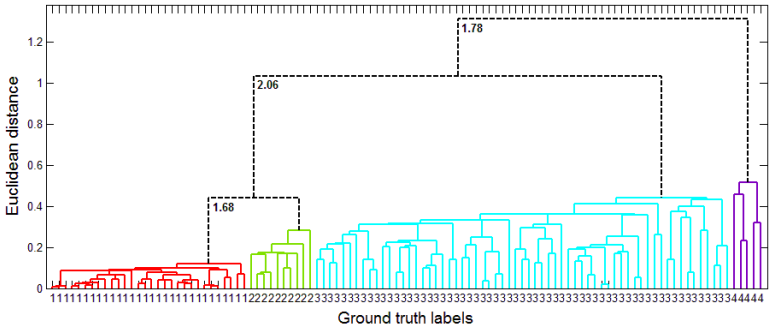
	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$	$\mathbf{C}_4$
SL + ZIC	0.68	0.61	0.72	0.73
UPGMA + ZIC	0.7	0.71	0.78	0.78
WPGMA + ZIC	0.65	0.69	0.77	0.77
UPGMC + ZIC	0.7	0.7	0.77	0.78

Table 3.1: $CPCC$ values for clustering solutions in Figure 3.4.

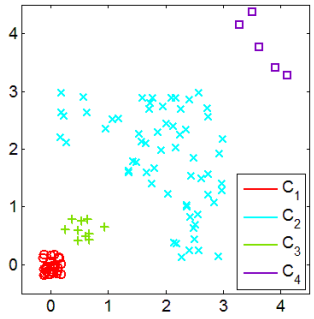
In general terms, although $CPCC$ values tend to be slightly lower for SL than for the rest of methods, these results (all values contained in the $[0.61, 0.78]$ interval) confirm that no high values of $CPCC$ (close to 1) should be expected unless the data fit some hierarchical structure. Moreover, the absence of significantly low $CPCC$ values (close to 0 or negative) indicates that the structure of the data within each cluster is reasonably well represented by the subdendrograms provided by the four methods.

There exist other different approaches to the issue of how to postprocess a dendrogram in order to obtain an optimal HPC solution (Rapoport and Fillenbaum, 1972; Lerman, 1981; Murtagh, 2007; Langfelder et al., 2008). Notwithstanding, they all suffer from the same two main problems: dendrogram-dependency (they all require as an input a dendrogram that somehow preserve the real clusters in its structure) and tuning-dependency (they all define parameter-dependent procedures whose performance is susceptible to an improper parameter tuning).

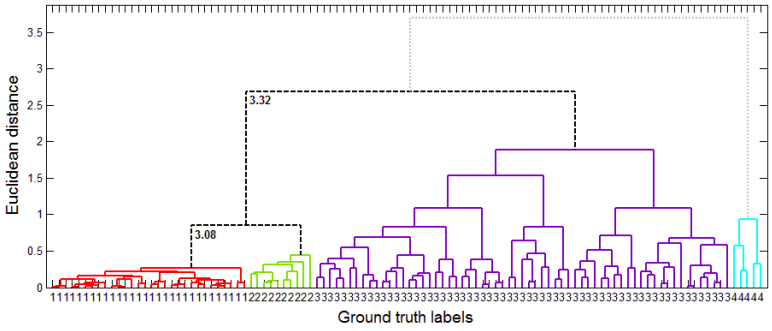
In fact, this two problems connect with some of the issues relative to the estimation of the optimal number of clusters in a dataset. Considering several of the approaches to this matter exposed in



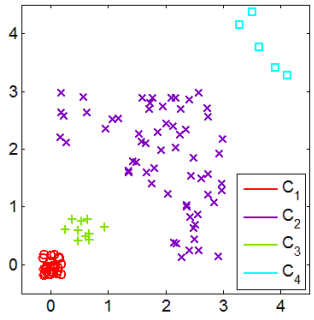
(a) SL + ZIC: Dendrogram ($\delta = 3$; $\iota_{th} = 1.65$).



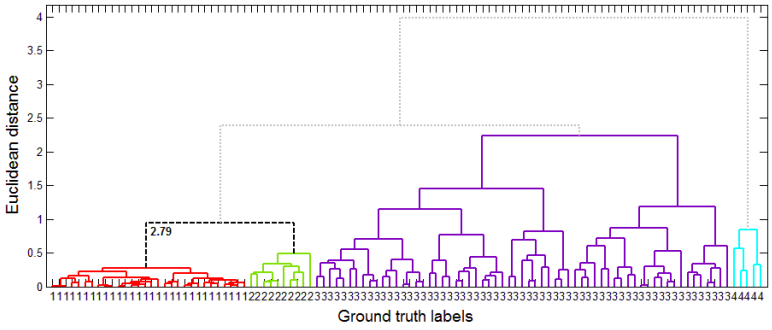
(b) SL + ZIC: HPC solution.



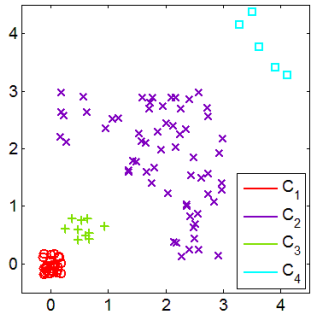
(c) UPGMA + ZIC: Dendrogram ($\delta = 4$; $\iota_{th} = 3.06$).



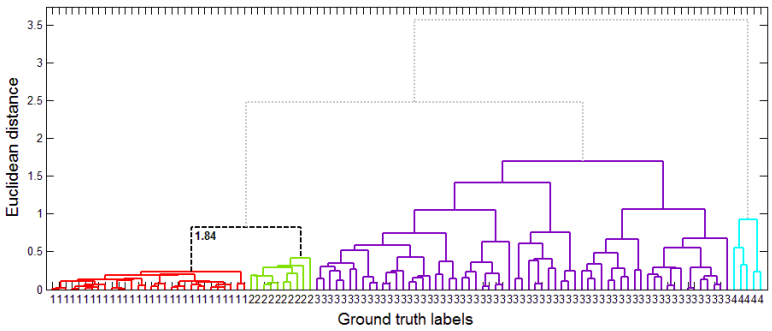
(d) UPGMA + ZIC: HPC solution.



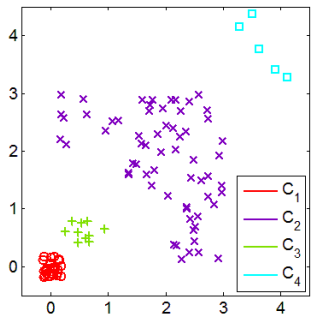
(e) WPGMA + ZIC: Dendrogram ($\delta = 5$; $\iota_{th} = 2.73$).



(f) WPGMA + ZIC: HPC solution.



(g) UPGMC + ZIC: Dendrogram ($\delta = 3$; $\iota_{th} = 1.83$).



(h) UPGMC + ZIC: HPC solution.

Figure 3.4: HC and HPC solutions on the *4toy* dataset by means of graph-based AHC methods combined with ZIC. ***(a) (c) (e) (g)*** The most inconsistent links in the dendrogram (black dashed links + inconsistency values), as well as higher links in the hierarchy (grey dotted links), are dismissed to obtain a 4-cluster HPC solution. Ground truth's cluster labels are indicated on the x-axis. ***(b) (d) (f) (h)*** Scatterplots: all HPC solutions match the ground truth 100% ($CI = 1$).

section 2.5.1, a dendrogram may seem to be an asset when it comes to determine the real value of K (Geva, 1999). However, while dendrograms may feasibly be convenient representations that allow identifying salient interrelationships and finding useful groups in a body of data, differing results can be easily provided by a single dendrogram depending on the scenario and the postprocessing strategy (Murtagh and Contreras, 2012), since they are both subordinate to open research questions such as how to estimate the number of clusters in the dataset or how to choose optimal cutting parameters (Langfelder et al., 2008).

In response to these limitations, proposals of parameter-free AHC adaptive algorithms arise with the aim of applying cluster merging and isolation criteria that have an impact on the structure of the dendrogram as it is being constructed, instead of postprocessing complete dendrograms.

3.2 Parameter-free AHC algorithms

Parameter-free AHC algorithms are proposed from an adaptive approach to clustering (see section 2.5.1), since they adaptively and dynamically adjust their behaviour in order to be flexible enough to provide optimal results in a diversity of clustering scenarios. Besides, their own nature is free of tunable parameters, so that their performance does not depend on user's criteria or prior expectations. This specific approach to clustering is constituted by the algorithms proposed by Fred and Leitão (2003) and Aidos and Fred (2011a), which are next detailed in sections 3.2.1 and 3.2.2, respectively.

3.2.1 AHC under a hypothesis of smooth dissimilarity increments

The problem of cluster defining criteria has been addressed in various forms (see section 2.3.2). Fred and Leitão (2000) propose a cluster isolation criterion based on a hypothesis of smooth dissimilarity increments between neighbouring objects within a cluster. Such a criterion is derived from the notions that *(i)* dissimilarity between objects within a cluster should not occur with abrupt changes and *(ii)* the merging of well-separated clusters incurs in abrupt changes in dissimilarity values.

From this starting point, let $\mathbf{X}$ be a dataset constituted by a set of N D-dimensional objects, let $\mathbf{x}_i$ be an arbitrary object of $\mathbf{X}$ and let $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k)$ be the triplet of nearest neighbours obtained as follows:

$$\begin{aligned} \mathbf{x}_j : j &= \arg\min_l \left\{ d_{\mathbf{x}_i \mathbf{x}_l} \right\}, \forall \mathbf{x}_l \in \mathbf{X} \mid l \neq i \\ \mathbf{x}_k : k &= \arg\min_l \left\{ d_{\mathbf{x}_j \mathbf{x}_l} \right\}, \forall \mathbf{x}_l \in \mathbf{X} \mid l \neq i, l \neq j \end{aligned} \tag{3.20}$$

where $d_{\mathbf{x}_i \mathbf{x}_j}$ expresses the proximity between objects $\mathbf{x}_i$ and $\mathbf{x}_j$ in terms of a dissimilarity measure.

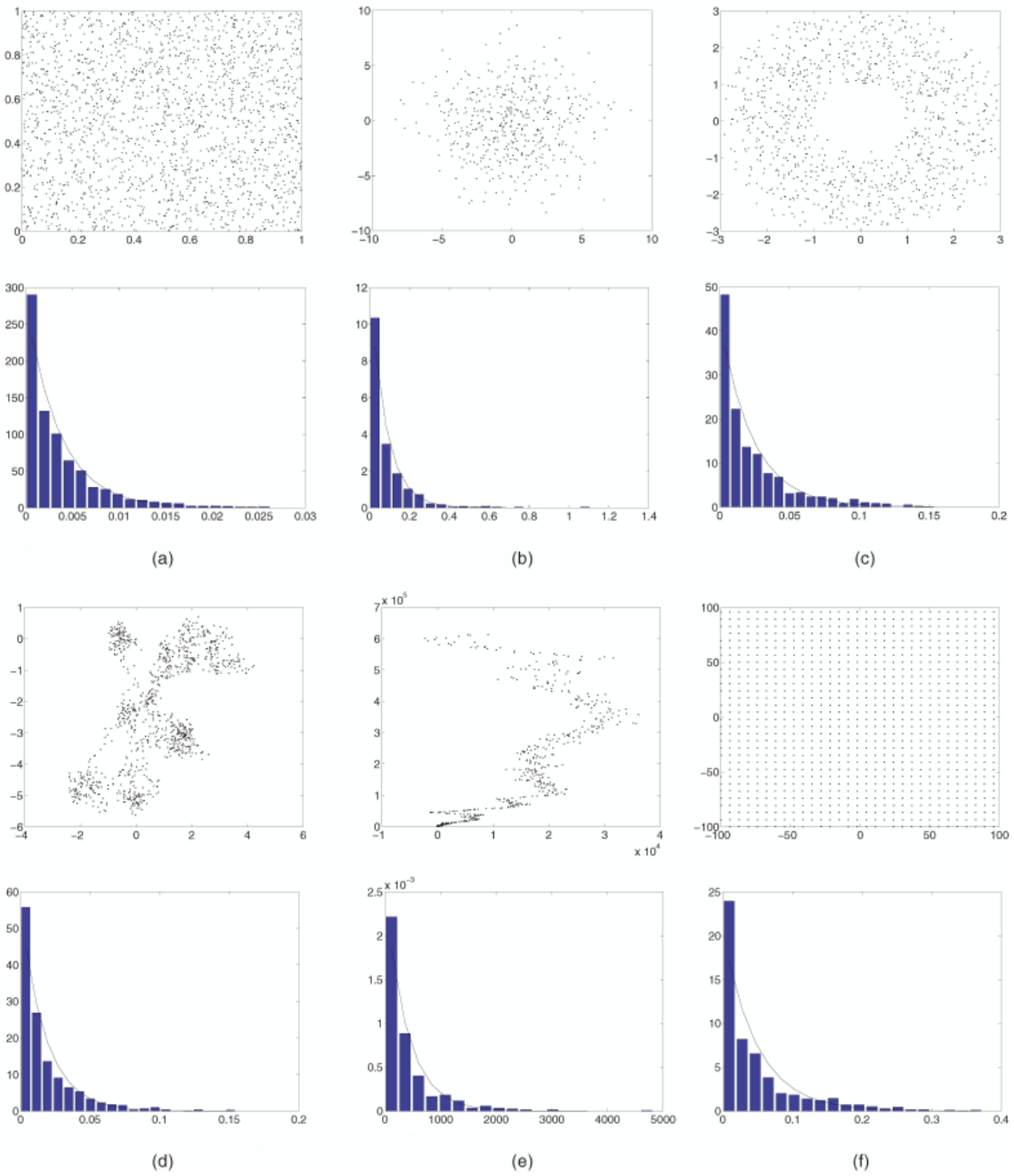


Figure 3.5: DID in different examples of clusters and data generation models (extracted from Fred and Leitão (2003)). Histograms (bar graphs) of the dissimilarity increments computed over neighbouring objects in the data approximately fit exponential functions (solid line curves). Examples include uniform, Gaussian, ring-shaped and noisy grid clusters, as well as other different stochastic data generation models. The dissimilarity measure used in all examples is the Euclidean distance.

Thus, the dissimilarity increment between neighbouring objects is defined as:

$$d_{inc}\left(\mathbf{x}_i,\, \mathbf{x}_j,\, \mathbf{x}_k\right) = \left| d_{\mathbf{x}_i\mathbf{x}_j} - d_{\mathbf{x}_j\mathbf{x}_k} \right| \tag{3.21}$$

which can be seen as the first derivative of the dissimilarity function at the first point of the list of neighbouring objects in $\mathbf{X}$ ordered according to equation 3.20.

Figure 3.5 shows empirical evidence on how the dissimilarity increments between neighbouring

objects within a cluster fit, regardless of the kind of cluster, a statistical distribution that, given its smooth evolution, can be approximated in terms of an exponential probability density function:

$$p(x) = \beta e^{-\beta x},\ x > 0 \tag{3.22}$$

where β is the rate parameter that defines the steepness of the dissimilarity increments distribution (DID). It is worth noting that distinct data generation models (*i.e.* clusters of different size, shape and distribution) lead to very similar curves, while variations in the data dispersion level (*i.e.* variations in the density of objects within the cluster) results in variations in the steepness of the DID, which can be simply modelled by varying the value of β.

Hence, according to this empirical hypothesis, a single parametric model (*i.e.* the exponential distribution defined in equation 3.22) can be utilised to characterise distinct kinds of clusters or data generation paradigms. In addition, when considering two well-separated clusters, it seems reasonable that dissimilarity increments between objects in different clusters are positioned far on the tail of the DID associated with the other cluster. This notion is explored by Fred and Leitão (2000) in order to define a cluster isolation criterion for AHC algorithms based on the SL method.

Since the nearest neighbour criterion (see equation 3.2) is the foundation of the SL method and the dissimilarity increments between nearest neighbours (see equation 3.21) are the basis of the model utilised to characterise clusters, Fred and Leitão (2000) propose to approximate dissimilarity increments by means of the gaps of the clusters (see Figure 3.6) in a SL-dendrogram. Thus, being $\mathbf{C}_i$ and $\mathbf{C}_j$ two clusters candidate to be merged into a new cluster $\mathbf{C}_k$ at some step in the agglomeration process, the gaps of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ (γ_i and γ_j, respectively) are defined as:

$$\begin{aligned} \gamma_i &= d_k - d_i = d_{ij} - d_i \\ \gamma_j &= d_k - d_i = d_{ij} - d_j \end{aligned} \tag{3.23}$$

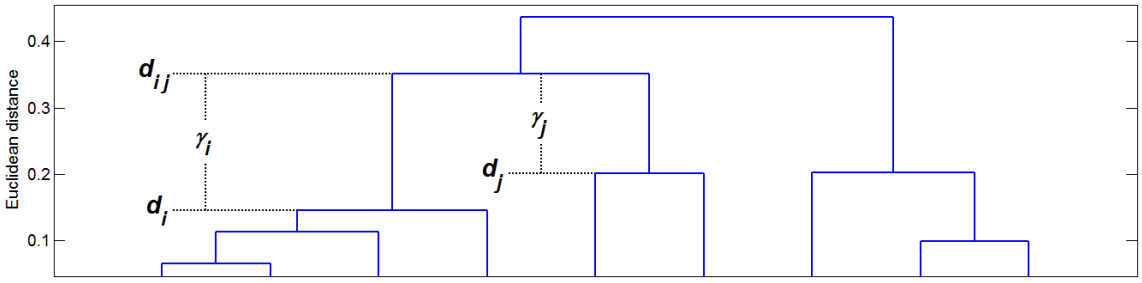


Figure 3.6: Gaps of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ in a SL-dendrogram.

According to this approximation, the statistical distribution of the gaps (*i.e.* the DID) within a cluster has a smooth evolution and should fit an exponential probability density function, whose data dispersion is characterised by its rate parameter β, as well as its mean value $\left(\mu = \frac{1}{\beta}\right)$. Furthermore, gaps between clusters that do not belong to the same cluster in the ground truth solution (*i.e.* increments computed for objects belonging to different real clusters) should have high values located on the tail of the DID of each cluster; that is to say, gaps generated by the merging of two

clusters that should not be merged are significantly higher than the mean value of the statistical distribution of the gaps within each cluster.

Therefore, an isolation criterion to decide whether two clusters merge or not in an AHC algorithm based on the SL method can be stated as (Fred and Leitão, 2000):

- Let $\mathbf{C}_i$ and $\mathbf{C}_j$ be two clusters candidate for merging, and γ_i and γ_j their respective gaps. Let μ_i and μ_j the mean values of the gaps distributions within $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively. If $\gamma_i > \alpha\mu_i$, cluster $\mathbf{C}_i$ is isolated and the agglomerative clustering process continue with the remaining objects; ditto on $\mathbf{C}_j$. If neither cluster exceeds the gap limit, $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster and the agglomerative clustering process continue.

It is worth noting that the isolation criterion situates the tail of the gaps distribution beyond a multiple (α) of its mean value. In order to obtain a clustering algorithm free of tunable parameters, Fred and Leitão (2003) propose the following setting: $\alpha = 3$.

Finally, since the estimation of the mean values of the gaps distribution may not be reliable for very small cluster sizes, the computation of the isolation criterion replaces the term $\alpha\mu_i$ by the following dynamic threshold (Fred and Leitão, 2003):

$$\gamma_{th}\left(\alpha, \mu_i, n_{\gamma i}, n_{\gamma j}\right) = \alpha\mu_i \, widen_{fact}\left(n_{\gamma i}, n_{\gamma j}\right) + delta_{fact}\left(n_{\gamma i}\right) \tag{3.24}$$

where $n_{\gamma i}$ and $n_{\gamma j}$ are the number of gaps within clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively (*i.e.* the sizes of the samples for the computation of μ_i and μ_j). The amplifying factor $widen_{fact}\left(n_{\gamma i}, n_{\gamma j}\right)$ is a monotonous decreasing function of $n_{\gamma i}$ and $n_{\gamma j}$, whose purpose is to enlarge the value of the threshold γ_{th} with the aim of compensating for possible underestimations of the true distribution mean when $n_{\gamma i}$ and $n_{\gamma j}$ are small and which is defined as:

$$widen_{fact}\left(n_{\gamma i}, n_{\gamma j}\right) = 1 + 3\left(1 - \frac{1}{1 + e^{-0.4(n_{\gamma i} - 10)}}\right)\left(2 - \frac{1}{1 + e^{-0.4(n_{\gamma j} - 10)}}\right) \tag{3.25}$$

In addition, since applying a multiplicative factor may not solve the underestimation problem of μ_i when $n_{\gamma i}$ is extremely low, the term $delta_{fact}\left(n_{\gamma i}\right)$ is added to the computation of the threshold γ_{th} in order to boost near zero estimates of μ_i for extremely small sized clusters:

$$delta_{fact}\left(n_{\gamma i}\right) = big_{val}\left(1 - \frac{1}{1 + e^{-10(n_{\gamma i} - 5)}}\right) \tag{3.26}$$

where big_{val} is a large positive number.

Thus, the isolation criterion defined according to the dynamic threshold in equation 3.24 gives rise to the first version of the SL-DID clustering algorithm (see Algorithm 3), which is a parameter-free AHC algorithm that combines elements from both graph-based and model-based approaches to clustering: graph-based, since it implements the SL method, and model-based, since it is defined by a cluster isolation criterion that models the DID in probabilistic terms. Moreover, since it is able to reject cluster unions, SL-DID is a more flexible algorithm than any implementation of the basic

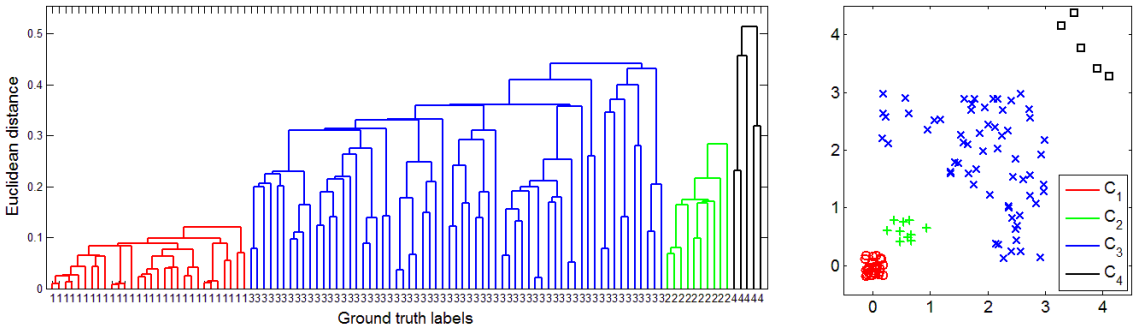
AHC method (see Algorithm 2). Despite this fact, its flexibility is certainly limited insomuch as a merging rejection involves the isolation of, at least, one of the two clusters candidate for merging (in the most flexible case, only one of the candidates is still able to participate in further unions, whereas the other is definitely isolated as one of the final clusters in the clustering solution).

As shown in Algorithm 3, SL-DID produces a data partitioning and simultaneous accessibility to the intrinsic data inter-relationships in terms of a dendrogram-type graph; *i.e.* SL-DID allows to automatically obtain a twofold clustering solution, since it simultaneously results in a HPC solution (represented by $\boldsymbol{\lambda}$) of a certain number of clusters (K) and in an AHC solution (represented

Algorithm 3 First version of the SL-DID algorithm (adapted from Fred and Leitão (2003)).

Input: Dataset $\mathbf{X}$ and parameter α ▷ Default setting: $\alpha = 3$
procedure
  $M_P : M_P(i,j) \leftarrow d_{\mathbf{x}_i\mathbf{x}_j},\ \forall\, \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X} \mid i \neq j$
  $\boldsymbol{\lambda} : \lambda_i \leftarrow i,\ \forall\, i \in \{1, N\}$ ▷ Initially, $\mathbf{P} = \{\mathbf{C}_1, ..., \mathbf{C}_N\} \mid \mathbf{C}_i = \{\mathbf{x}_i\}$
  $\boldsymbol{\Delta} : \overline{\Delta}_i \leftarrow [0\ 0\ 0],\ \forall\, i \in \{1, N-1\}$
  $d_i \leftarrow 0,\ n_{\gamma i} \leftarrow 0,\ \mu_i \leftarrow 0,\ \forall\, i \in \{1, N\}$
  $k \leftarrow N+1$
  while $M_P \neq \emptyset$ **do**
    $(m,p) \leftarrow \arg\min_{(q,r)} \{M_P(q,r)\},\ d_{ij} \leftarrow \min\{M_P\}$
    $i \leftarrow \lambda_m,\ j \leftarrow \lambda_p$
    $\gamma_i \leftarrow d_{ij} - d_i,\ \gamma_j \leftarrow d_{ij} - d_j$
    if $(\gamma_i < \gamma_{th}(\alpha, \mu_i, n_{\gamma i}, n_{\gamma j})) \wedge (\gamma_j < \gamma_{th}(\alpha, \mu_j, n_{\gamma j}, n_{\gamma i}))$ **then**
      $\lambda_l \leftarrow k,\ \forall\, l \in \{1, N\} \mid \lambda_l = i,\ \lambda_l = j$ ▷ $\mathbf{C}_k = \mathbf{C}_i \cup \mathbf{C}_j$
      $\overline{\Delta}_{(k-N)} \leftarrow [\, i\ j\ d_{ij}]$
      $d_k \leftarrow d_{ij},\ n_{\gamma k} \leftarrow n_{\gamma i} + n_{\gamma j} + 2,\ \mu_k \leftarrow \frac{n_{\gamma i}}{n_{\gamma k}}\mu_i + \frac{n_{\gamma j}}{n_{\gamma k}}\mu_j + \frac{1}{n_{\gamma k}}(\gamma_i + \gamma_j)$
      $k \leftarrow k+1$
      Update M_P: Dismiss the proximities between objects belonging to $\mathbf{C}_i$ and $\mathbf{C}_j$
    else
      if $(\gamma_i \geq \gamma_{th}(\alpha, \mu_i, n_{\gamma i}, n_{\gamma j}))$ **then**
        Update M_P: Dismiss all the proximities relative to objects belonging to $\mathbf{C}_i$
      end if
      if $(\gamma_j \geq \gamma_{th}(\alpha, \mu_j, n_{\gamma j}, n_{\gamma i}))$ **then**
        Update M_P: Dismiss all the proximities relative to objects belonging to $\mathbf{C}_j$
      end if
    end if
  end while
end procedure
Output: Label vector $\boldsymbol{\lambda}$ and dendrogram $\boldsymbol{\Delta}$

by $\boldsymbol{\Delta}$) composed of K dendrograms, each one of which describes the internal structure of a single hard cluster; $\boldsymbol{\Delta}$ actually includes K independent dendrograms and its lasts $K-1$ rows are empty $\left(\overline{\Delta}_i = [0\ 0\ 0]\,, \forall\, i \in \{N-K+1, N-1\}\right)$. This fact is illustrated in Figure 3.7, which shows the final clustering solution provided by SL-DID algorithm on the *4toy* dataset previously used in Figures 3.3 and 3.4.



(a) SL-DID: AHC solution. ***(b)*** SL-DID: HPC solution.

Figure 3.7: AHC and HPC solutions on the *4toy* dataset by means of SL-DID algorithm. ***(a)*** The AHC solution comprises four dendrograms, one for every cluster in the HPC solution. ***(b)*** The HPC solution matches the ground truth 100% ($CI = 1$).

It is worth noting the insight into the nature of the dataset gained from this twofold clustering solution. Apart from the fact that the optimal HPC solution is reached (it equals the ground truth), the absence of significantly low CPCC values (see Table 3.2) indicates that internal structure of clusters is properly represented by the resultant dendrograms, even though clusters are not hierarchically structured and therefore high CPCC values are not obtained either.

$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$	$\mathbf{C}_4$
0.68	0.72	0.61	0.73

Table 3.2: $CPCC$ values for dendrograms in Figure 3.7a.

Thus, the dendrograms provide useful information on the constitution (size, density and some notions on the objects distribution) of the clusters: *e.g.* whereas $\mathbf{C}_1$ is the most dense and compact cluster, $\mathbf{C}_3$ is the most populated and $\mathbf{C}_4$ is the smallest and sparsest cluster in the dataset. In addition, dendrograms allows to identify and contextualise each individual object within its cluster: core and frontier objects can be located, as well as neighboring objects, potential central representatives or possible outliers.

Concerning the computational requirements, the behaviour of SL-DID is $O\left(N^2\right)$ in storage terms and $O\left(N^2 \log N\right)$ in time terms:

- As stated in section 2.2, the computation of M_P (see line 3 in Algorithm 3) requires the storage of $\frac{N^2-N}{2}$ different proximity values. In addition, whereas $\boldsymbol{\lambda}$ and $\boldsymbol{\Delta}$ respectively require to store N and $3\,(N-1)$ values (see lines 4 and 5), the storage requirements of the rest of va-

riables (see line 6) vary in the range $[N, 2N-1]$ (see line 15), since their size depends on the number of hard clusters the algorithm identifies (K), which may vary in the range $[1, N]$. Therefore, due to the requirements of M_P, SL-DID is an $O(N^2)$ algorithm in storage terms.

- While the initialisation of the rest of variables linearly depends on N, $\frac{N^2-N}{2}$ proximities have to be calculated to obtain M_P, which involves a N^2-dependent computation time. However, the minimum value of M_P needs to be sought at every stage of the agglomeration process (see line 9). In order to optimise this step, the values in M_P can be previously sorted by means of $O(n \log n)$ sorting algorithms such as Quicksort, Mergesort or Heapsort, being n the number of elements to be sorted (Knuth, 1998). Since SL-DID requires to sort $\frac{N^2-N}{2}$ proximities, this sorting step involves a $(N^2 \log N)$-dependent computation time.

  As for the rest of the agglomeration process, it is not finished while M_P is not empty (see line 8). Considering that the proximities between objects belonging to $\mathbf{C}_i$ and $\mathbf{C}_j$ are always dismissed regardless of whether they merge into a new cluster or not (see lines 17, 20 and 23), the maximum possible number of agglomeration stages is $N-1$, since no more than $N-1$ pairs of clusters can arise as candidates for merging (see line 4 in Algorithm 2). Apart from the updating of M_P (it involves $\frac{N^2-N}{2}$ dismissing operations in total, which leads to a N^2-dependent computation time), the rest of operations performed at every agglomeration stage (see lines 9–16, 19 and 22) are independent from N, which, given that no more than $N-1$ stages are possible, involves a N-dependent computation time.

  Therefore, due to the requirements of the sorting step, SL-DID is an $O(N^2 \log N)$ algorithm in time terms, although this behaviour is usually referred as $O(N^2)$, since the $\log N$ factor is negligible in front of the N^2 factor when N tends to be large.

Finally, Fred and Leitão (2003) provide a set of experimental results that illustrate the performance of SL-DID algorithm on a diversity of both synthetic and real datasets and in comparison with other clustering algorithms. SL-DID prove to be a highly versatile algorithm in identifying clusters of different sizes, shapes, densities and distributions, intrinsically finding the number of clusters without the requirement of any user intervention. In general terms, interesting results are obtained when clustering synthetic datasets, whereas the performance of SL-DID significantly worsens when clustering both synthetic datasets with touching or overlapped clusters of similar densities and real datasets –which also include touching and overlapped clusters– Fred and Leitão (2003, Section 5, pages 951–954). The cluster isolation criterion SL-DID is based on is the reason for this behaviour, since it rejects abrupt dissimilarity changes (*i.e.* large gaps) to avoid the merging of well-separated clusters. If two clusters with a similar density of objects are touching or overlapped enough to not incur in a significantly large gap, the criterion is not able to isolated them separately, being this the main limitation of SL-DID in terms of performance.

3.2.2 AHC based on high order dissimilarities

While it gives rise to a highly versatile parameter-free AHC algorithm (SL-DID), the estimation of the DID between neighboring objects in terms of an exponential function proposed by Fred and Leitão (2000) is just an empirical approximation. In order to obtain a more accurate estimation, Aidos and Fred (2011b) analytically derive a general statistical model for the DID under milder approximations.

Assuming that data come from a cluster with a 2-dimensional Gaussian distribution as an underlying hypothesis, the DID between neighboring objects within a cluster can be analytically expressed as (Aidos and Fred, 2011b):

$$p(\omega;\mu) = \frac{\pi\left(2-\sqrt{2}\right)^2}{4\mu^2}\,\omega\, e^{\left(-\frac{\pi(2-\sqrt{2})^2}{4\mu^2}\omega^2\right)} + \frac{\pi^2\left(2-\sqrt{2}\right)^3}{8\sqrt{2}\mu^3}\left(\frac{4\mu^2}{\pi\left(2-\sqrt{2}\right)^2}-\omega^2\right) e^{\left(-\frac{\pi(2-\sqrt{2})^2}{8\mu^2}\omega^2\right)} erfc\left(\frac{\sqrt{\pi}\left(2-\sqrt{2}\right)}{2\sqrt{2}\mu}\omega\right) \tag{3.27}$$

where ω is an stochastic variable that represents the values of the dissimilarity increments within the cluster, μ is its first moment ($\mu = E[\omega]$) –*i.e.* the mean value of the dissimilarity increments– and $erfc$ is the complementary error function (Andrews, 1998):

$$erfc(x) = \frac{2}{\sqrt{\pi}}\int_x^{+\infty} e^{-t^2}\,dt \tag{3.28}$$

It is worth noting that, in spite of having assumed a 2-dimensional Gaussian distributed cluster, this approximation for the DID only depends on ω and μ, so that it can be applied to other higher-dimensional cluster distributions –Aidos and Fred (2011b) provide empirical evidence that confirms the possibility of this generalisation–.

Thus, Aidos and Fred (2011a) propose to incorporate this analytical model for the DID in a second version of the SL-DID algorithm –it has been also applied in the design of a HPC algorithm (Aidos and Fred, 2011c)–, whose schematic description is shown in Algorithm 4. This new version presents several differences in comparison with Algorithm 3, which are next summarised:

- Small clusters receive a different treatment than the rest (see lines 6–19 in Algorithm 4, being N_i and N_j the number of objects in clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively). According to Aidos and Fred (2011a), 6 is the minimum number of objects a cluster has to contain in order that a rough estimation of its DID may be computed. Therefore, if both candidate clusters have less than 6 objects, they automatically merge into a new cluster. Furthermore, if only one of the candidates has less than 6 objects and the mean value of its gaps distribution (μ_i and μ_j are the mean values of the gaps distributions of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively) does not fall in the tail of the DID of the other candidate, both clusters also merge.
- The merging of large enough clusters (whose number of objects is greater than 6) is subject

to a test similar to the isolation criterion the Algorithm 3 is based on (see lines 21–27, being γ_i and γ_j the gaps of clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively): those clusters candidate whose gap falls in the tail of the DID of the other candidate are definitely isolated for the rest of the agglomeration process.

- In case none of the two clusters candidate for merging is isolated, an MDL criterion is applied in order to determine whether these two clusters finally merged or not (see lines 28–34). Being $p(\omega;\mu)$ the DID stated in equation 3.27, the description length for cluster $\mathbf{C}_i$ is defined as:

$$DL(\mathbf{C}_i) = \frac{1}{2}(1 - \log(12)) + \log(\mu_i) + \frac{1}{2}\log I(\mu_i) - \log(p(\omega_i;\mu_i)) \tag{3.29}$$

where $I(\cdot)$ is the expected Fisher information $\left(I(\mu_i) = -E\left[\frac{\partial^2 p(\omega_i;\mu_i)}{\partial^2 \mu_i}\right]\right)$. Thus, the resultant MDL criterion is that defined in line 29.

- Algorithm 4 is certainly more flexible than Algorithm 3, since it allows to reject possible cluster unions without necessarily involving the definitive isolation of, at least, one of the two clusters candidate for merging (see lines 12, 18 and 33), so that they are still able to merge with other clusters at further stages of the agglomeration process.

Algorithm 4 Schematic description of the second version of the SL-DID algorithm (extracted from Aidos and Fred (2011a)).

1: *Input*: Dataset $\mathbf{X}$
2: **procedure**
3: Each object is a cluster.
4: **repeat**
5: The most similar pair of clusters not yet tested $(\mathbf{C}_i, \mathbf{C}_j)$ is chosen.
6: **if** $(N_i < 6) \wedge (N_j < 6)$ **then**
7: $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster.
8: **else if** $(N_i \geq 6) \wedge (N_j < 6)$ **then**
9: **if** $(\mu_j < 7\mu_i)$ **then**
10: $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster.
11: **else**
12: $\mathbf{C}_i$ and $\mathbf{C}_j$ do not merge into a new cluster.
13: **end if**
14: **else if** $(N_i < 6) \wedge (N_j \geq 6)$ **then**
15: **if** $(\mu_i < 7\mu_j)$ **then**
16: $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster.
17: **else**
18: $\mathbf{C}_i$ and $\mathbf{C}_j$ do not merge into a new cluster.
19: **end if**
20: **else**

γ_i and γ_j are computed.
if γ_i is in the tail of $\mathbf{C}_j$ **then**
$\mathbf{C}_i$ is isolated.
end if
if γ_j is in the tail of $\mathbf{C}_i$ **then**
$\mathbf{C}_j$ is isolated.
end if
if γ_i is not in the tail of $\mathbf{C}_j$ and γ_j is not in the tail of $\mathbf{C}_i$ **then**
$DL(\mathbf{C}_i)$, $DL(\mathbf{C}_j)$ and $DL(\mathbf{C}_i \cup \mathbf{C}_j)$ are computed.
if $(DL(\mathbf{C}_i \cup \mathbf{C}_j) \leq DL(\mathbf{C}_i) + DL(\mathbf{C}_j))$ **then**
$\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster.
else
$\mathbf{C}_i$ and $\mathbf{C}_j$ do not merge into a new cluster.
end if
end if
end if
until All pairs of clusters should not be merged.
end procedure
Output: Label vector $\boldsymbol{\lambda}$ and dendrogram $\boldsymbol{\Delta}$

Concerning the computational requirements, Algorithm 4 behaves similarly to Algorithm 3; *i.e.* it is $O(N^2)$ in storage terms, and $O(N^2 \log N)$ in time terms (Aidos and Fred, 2011a).

Finally, experimental results show that the performance of both versions of SL-DID algorithm is equivalent. While synthetic clusters of different sizes, shapes, densities and distributions are properly identified in a parameter-free environment, both synthetic datasets with touching or overlapped clusters of similar densities and real datasets cause a significant decrease in the performance of both versions of the algorithm (Aidos and Fred, 2011a, Section 4.3, Table 2, page 291).

3.3 Discussion

While postprocessing dendrograms proceeded from basic AHC methods has proved to be an insufficient strategy in order to guarantee a proper estimation of the real number of clusters in a dataset (see section 3.1.3), parameter-free AHC algorithms resulting from combining graph-based (SL methods) and model-based (probabilistic modelling of the nature of clusters) approaches to clustering seem to be a more proper choice to such a task.

In addition, algorithms proposed by Fred and Leitão (2003) and Aidos and Fred (2011a) suitably fit the characteristics of the scenario arisen from posing the modelling of learners' activity in on-

line discussion forums as a clustering problem, since they have proved to be versatile enough to identify clusters of different nature and characteristics (see sections 3.2.1 and 3.2.2). The twofold clustering solution (both partitional and hierarchical) they result in by means of a free of tunable parameters procedure allows to automatically determine the number of clusters, to perform a visual exploration of the data and to contextualise each individual object within its cluster. However, the scope of application of these algorithms is limited by its lack of good results when dealing with datasets that present touching or overlapped clusters of similar density, which can be a serious drawback in highly context-dependent clustering scenarios.

In this context and with the purposes of both outperforming the previous algorithms of its class and being a more suitable choice to model learners' activity in online discussion forums, the main contributions of the present thesis are presented and described in Chapter 4: the design of a novel parameter-free AHC algorithm based on the definition of two new cluster merging criteria.

Chapter 4

A novel parameter-free AHC algorithm based on two new cluster merging criteria

As aforementioned in the previous chapter, parameter-free AHC algorithms proposed so far in the literature lack good performance results when handling touching or overlapped clusters of similar density. This limitation seems to be caused by the nature of the cluster isolation criterion they are based on (see section 3.2 for further details). Thus, possible ways to improve this deficiency might be both to enhance the existing criterion and to combine different criteria with the aim of compensating their respective lacks. Therefore, with the goal of overcoming these deficiencies and improving the current approach to parameter-free AHC, this chapter is focused on presenting the main contributions of the present thesis, which can be summarised into the definition of two new cluster merging criteria and the design of a novel parameter-free AHC algorithm derived from these criteria, whose characteristics are analysed both in comparison with previous parameter-free AHC algorithms and in terms of its computational requirements.

For that purpose, this fourth chapter is structured as follows. Two new cluster merging criteria are defined and presented in section 4.1; a novel parameter-free AHC algorithm derived from the proposed criteria is implemented and described in detail in section 4.2; the computational requirements, both in terms of storage and time, of the proposed algorithm are analysed in section 4.3; and, finally, the main differences and significant improvements of the new algorithm with respect to its predecessors are discussed in section 4.4.

4.1 Beyond the cluster isolation criterion based on dissimilarity increments

Parameter-free AHC algorithms existing to date derive from the cluster isolation criterion stated by Fred and Leitão (2000), which is based on a model of the dissimilarity increments distribution (DID) between neighbouring objects (see section 3.2). Nonetheless, despite their interesting features and very promising results, both versions of the SL-DID algorithm derived from this cluster isolation criterion (Fred and Leitão, 2003; Aidos and Fred, 2011a) present problems for identifying both touching and overlapped clusters (see sections 3.2.1 and 3.2.2).

The reasons for these limitations can be found in the initial notions the cluster isolation criterion proposed by Fred and Leitão (2000) is based on, which states that *(i)* dissimilarity between objects within a cluster should not occur with abrupt changes and that *(ii)* the merging of well-separated clusters incurs in abrupt changes in dissimilarity values. Whereas the first statement leads to the ability of the probabilistic model derived from this criterion to successfully fit clusters of all kind, the second statement clearly indicates the reasons for the lack of performance of both versions of SL-DID algorithm: since both touching and, specially, overlapped clusters are not well-separated, they do not provoke abrupt changes in dissimilarity values and they cannot therefore be isolated by the criterion.

Hence, with the aim of overcoming these limitations, it is at this point where the main contribution of the present thesis arises, which can be defined as the **implementation of a novel parameter-free AHC algorithm** –whose full design is described and analysed in section 4.2– that improves the abilities of SL-DID algorithm by satisfying the following premises:

- Maintaining the upsides of SL-DID algorithm, which can be mainly summarised into having a behaviour free of tunable parameters, generating twofold clustering solutions and being able to handle clusters of any kind.
- Gaining the ability of successfully dealing with touching and overlapped clusters.
- Not involving a significant increase of the computational requirements in comparison to SL-DID algorithm, whose behaviour is $O(N^2)$ in storage terms and $O(N^2 \log N)$ in time terms.
- And, as a consequence of the previous premises, having a wider scope of application than that of SL-DID algorithm (see Chapter 5 for further details).

Thus, in order to achieve these improvements, this novel algorithm bases the decision about whether a pair of clusters merge or not on two different new cluster merging criteria, whose definitions are main contributions of the present thesis as well. While the first new cluster merging criterion

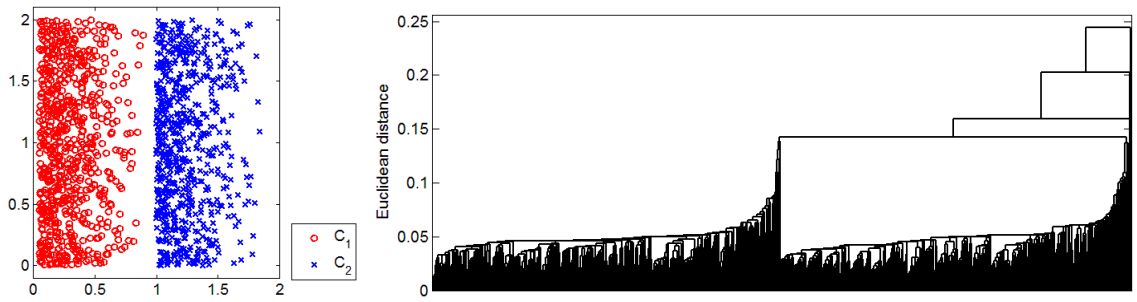
can be seen as an improved version of the cluster isolation criterion proposed by Fred and Leitão (2000), the second new cluster merging criterion is put forward to compensate for the main limitation of criteria exclusively based on the DID:

1. **The LSS criterion** (LSS refers to Local Standard Score): a cluster merging local criterion whose main difference from the cluster isolation criterion proposed by Fred and Leitão (2000) –whilst being directly derived from it– resides in the local and density-dependent nature of its threshold decision, which improves the behaviour of the criterion in the frontier regions between barely-separated, or even touching, clusters (see section 4.1.1 for further details).

2. **The GCSS criterion** (GCSS refers to Global Cumulative Standard Score): a cluster merging global criterion which, based on the progressive increment of the cumulative proximity levels within each cluster throughout the agglomeration process –which do incur in abrupt changes when clusters, well-separated or not, merge–, allows to avoid possible erroneous cluster unions between both touching and overlapped clusters (see section 4.1.2 for further details).

4.1.1 The LSS cluster merging criterion

In order both to gain an insight into the downsides of the cluster isolation criterion proposed by Fred and Leitão (2000) and to understand the improvement a local cluster merging criterion may involve, let the *2bars* dataset shown in Figure 4.1a be considered (see section 5.2.2.3 for further details).



(a) Scatterplot of the *2bars* dataset.

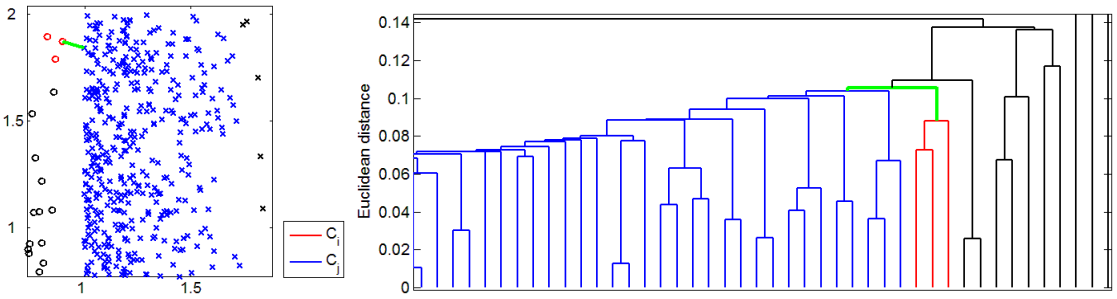
(b) *2bars* dataset: SL-dendrogram.

Figure 4.1: The *2bars* dataset.

It is a 2-dimensional synthetic dataset consisting of two nearly balanced clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) whose density of objects progressively diminishes. Such a distribution gives rise to a frontier between clusters delimited by a high-density region and a low-density region, each one belonging to a different cluster. This dataset has been previously used in the literature with the aim of exposing some limitations of the basic AHC algorithm based on the SL method (Aidos and Fred, 2011a, Section 3.3, Figure 4, page 288). Thus, the resultant SL-dendrogram on the *2bars* dataset is shown in Figure 4.1b.

At a certain stage of the agglomeration process, a pair of clusters candidate for merging ($\mathbf{C}_i$ and $\mathbf{C}_j$) arises. Being two remarkably unbalanced clusters ($N_i = 3$, $N_j = 666$), they both merge into a new cluster in the dendrogram ($\mathbf{C}_k$) as shown in Figure 4.2. This precise merging presents several peculiarities:

- $\mathbf{C}_i$ is entirely composed by objects belonging to $\mathbf{C}_1$, whereas $\mathbf{C}_j$ is constituted only by objects belonging to $\mathbf{C}_2$; *i.e.* $\mathbf{C}_i \subset \mathbf{C}_1$ and $\mathbf{C}_j \subset \mathbf{C}_2$. Thus, **it is a troublesome merging**, since it results in a new cluster composed of objects belonging to different clusters in the ground truth, and algorithms like SL-DID or the one proposed in the present chapter **should therefore reject it**.
- The merging takes place before both $\mathbf{C}_1$ and $\mathbf{C}_2$ are completely formed in the dendrogram and, as shown in Figure 4.2b, it does not cause any significantly high gap with respect to the gaps already existing within $\mathbf{C}_i$ or $\mathbf{C}_j$.
- The neighbouring objects ($\mathbf{x}_m \in \mathbf{C}_i$ and $\mathbf{x}_p \in \mathbf{C}_j$) that give rise to the merging belong to very different regions in density terms: $\mathbf{x}_m$ belongs to the lowest density region in $\mathbf{C}_1$, whereas $\mathbf{x}_p$ belongs to the highest density region in $\mathbf{C}_2$.



(a) Candidate clusters $\mathbf{C}_i$ and $\mathbf{C}_j$.

(b) $\mathbf{C}_i$ and $\mathbf{C}_j$ in the SL-dendrogram.

Figure 4.2: *2bars* dataset: detail of the agglomeration process. ***(a)*** Zoom in on the scatterplot in Figure 4.1a. Candidate clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ are depicted by colours (see legend); the proximity between neighbouring objects is green-coloured. ***(b)*** Zoom in on the dendrogram in Figure 4.1b. $\mathbf{C}_i$ and $\mathbf{C}_j$ can be identified according to their respective colours in Figure 4.2a; the link resulting from the merging is green-coloured.

Considering the perspective of the cluster isolation criterion the SL-DID algorithm is based on, this particular merging presents two problems that prevent the algorithm from accurately identifying $\mathbf{C}_1$ and $\mathbf{C}_2$ (the real clusters present in the ground truth of the dataset):

- Despite of being a non-desirable merging, it cannot be rejected by the cluster isolation criterion proposed by Fred and Leitão (2000), since **neither of the generated gaps falls in the tail of the DIDs of $\mathbf{C}_i$ and $\mathbf{C}_j$**, respectively.
- According to the behaviour of the SL-DID algorithm in its first version (its second version is certainly more flexible), if the merging were hypothetically rejected, at least one of the

candidate clusters ($\mathbf{C}_i$, $\mathbf{C}_j$, or both) **would keep isolated** for the rest of the agglomeration process, which would involve that at least one of the clusters in the ground truth ($\mathbf{C}_1$, $\mathbf{C}_2$ or both) **could not be properly identified** by the algorithm.

Nevertheless, from a local perspective, it can be observed that the proximity between neighbouring objects $\mathbf{x}_m$ and $\mathbf{x}_p$ (*i.e.* the proximity between clusters $\mathbf{C}_i$ and $\mathbf{C}_j$) is significantly high in comparison with **the proximities between objects existing in the vicinities of $\mathbf{x}_m$ and, specially,** $\mathbf{x}_p$, which belongs to the high-density region in cluster $\mathbf{C}_j$.

Thus, with the aim of increasing the robustness of the clustering process, as well as to improve its flexibility in case the merging between candidate clusters is rejected, the **LSS criterion** is defined under the following premises:

1. The union between candidate clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ will be rejected if any of the gaps caused by the merging falls in the tail of the DIDs existing **in the vicinities of neighbouring objects $\mathbf{x}_m$ and $\mathbf{x}_p$.** In this way, and continuing with the *2bars* dataset example, the merging between candidate clusters would be rejected from the point of view of $\mathbf{C}_j$, since its gap would fall in the tail of the DID present in the vicinity of $\mathbf{x}_p$, which is composed of objects much nearer to $\mathbf{x}_p$ than to $\mathbf{x}_m$.
2. In case of rejection, any of the candidate clusters **will not be isolated**. Since the merging criterion operates in local terms, the nature of the rejection will be considered local as well. Hence, in case of local rejection, candidate clusters will remain separated and **pairwise proximities between objects belonging to the vicinities of $\mathbf{x}_m$ and $\mathbf{x}_p$** will be dismissed from the proximity matrix of the dataset. In this way, $\mathbf{C}_i$ and $\mathbf{C}_j$ would not merge through this local region (defined by $\mathbf{x}_m$, $\mathbf{x}_p$ and their respective vicinities), but there would remain the possibility that they merge in further stages of the agglomeration process through a different local region (defined by a different pair of neighbouring objects and their respective vicinities).

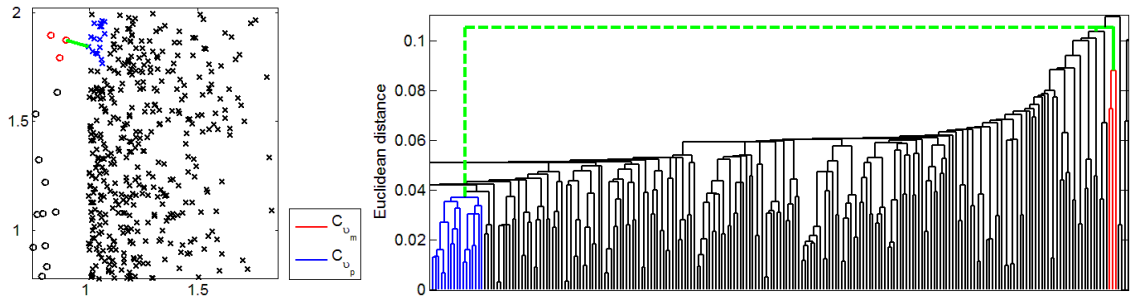
Clearly, these premises require of the LSS criterion both to identify the vicinities of the neighbouring objects and to estimate their DIDs at every stage of the agglomeration process. Apart from the need of a proper definition of vicinity, such a requirement may certainly increase the computational cost of the clustering process. Therefore, the following question arises:

- How the vicinity of every object in the dataset can be defined during the agglomerative construction of the dendrogram without involving a significant increase of the computational cost of the clustering process?

Considering that the vicinity of an object has to be necessarily defined by the nearest neighbours of the object in the dataset, it can be properly estimated from the hierarchy of clusters represented in the dendrogram, since an SL-dendrogram defines clusters of nearest neighbouring objects.

Hence, the LSS criterion establishes that the vicinity of an object $\mathbf{x}_i$ is defined by that cluster $\mathbf{C}_{v_i}$ which is **the smallest cluster in the dendrogram that contains** $\mathbf{x}_i$ **and whose size is greater than or equal to a certain threshold size** $(N_{v_i} \geq N_{MIN})$ –*i.e.* the smallest cluster in the dendrogram that contains a minimum amount of objects, including $\mathbf{x}_i$–.

Such a criterion is illustrated in Figure 4.3, where, continuing with the *2bars* dataset example, the vicinity of $\mathbf{x}_p$ is defined by cluster $\mathbf{C}_{v_p}$ (*i.e.* that subcluster of $\mathbf{C}_j$ which contains a minimum amount of objects, including $\mathbf{x}_p$). As shown in Figure 4.3b, it is worth noting the significant difference existent between the gap resulting from the merging when considering $\mathbf{C}_j$ and the gap resulting from the merging when considering $\mathbf{C}_{v_p}$.



(a) Vicinities of $\mathbf{x}_m$ and $\mathbf{x}_p$.

(b) $\mathbf{C}_{v_m}$ and $\mathbf{C}_{v_p}$ in the SL-dendrogram.

Figure 4.3: *2bars* dataset: detail of the vicinities of two neighbouring objects. ***(a)*** Zoom in on the scatterplot in Figure 4.1a. The vicinities of $\mathbf{x}_m$ and $\mathbf{x}_p$ are clusters $\mathbf{C}_{v_m}$ and $\mathbf{C}_{v_p}$, respectively, which are depicted by colours (see legend); the proximity between neighbouring objects is green-coloured. ***(b)*** Zoom in on the dendrogram in Figure 4.1b. $\mathbf{C}_{v_m}$ and $\mathbf{C}_{v_p}$ can be identified according to Figure 4.3a; differences between the resultant gaps considering $\mathbf{C}_j$ (green link) and $\mathbf{C}_{v_p}$ (green dashed link) are illustrated.

Thus, let $\mathbf{X}$ be a dataset constituted by a set of N D-dimensional objects. Given an agglomeration process of the objects in $\mathbf{X}$ based on the SL method, let $\mathbf{x}_m$ and $\mathbf{x}_p$ be a pair of neighbouring objects in $\mathbf{X}$ that give rise to a pair clusters ($\mathbf{C}_i$ and $\mathbf{C}_j$) candidate for merging ($\mathbf{x}_m \in \mathbf{C}_i$ and $\mathbf{x}_p \in \mathbf{C}_j$). Finally, let d_{ij} be the proximity between clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ according to the SL method $\left(d_{ij} = d_{\mathbf{x}_m \mathbf{x}_p}\right)$. The LSS criterion determines whether $\mathbf{C}_i$ and $\mathbf{C}_j$ should merge into a new cluster or not and it is defined as follows.

Firstly, let $\mathbf{C}_x$ be any given cluster in the dendrogram $\mathbf{\Delta}$ resulting from the agglomeration process of the objects in $\mathbf{X}$ and let $\overline{\gamma}_x$ be the sample consisting of the gaps nested within $\mathbf{C}_x$. **The standard score statistic of cluster** $\mathbf{C}_x$ (ss_x) is defined as the standard score of the largest gap within $\mathbf{C}_x$ (Γ_x) with respect to $\overline{\gamma}_x$:

$$ss_x = \frac{\Gamma_x - \mu_x}{\sqrt{\sigma_x - \mu_x^2}} \tag{4.1}$$

where $\Gamma_x = \max\{\overline{\gamma}_x\}$, and μ_x and σ_x are the first and second moments of $\overline{\gamma}_x$, respectively:

$$\mu_x = \frac{1}{n_{\gamma x}} \sum_{l=1}^{n_{\gamma x}} \overline{\gamma}_{x_l} \quad , \quad \sigma_x = \frac{1}{n_{\gamma x}} \sum_{l=1}^{n_{\gamma x}} \overline{\gamma}_{x_l}^2 \tag{4.2}$$

being $\overline{\gamma}_{x_l}$ and $n_{\gamma x}$ the lth observation and the length of sample $\overline{\gamma}_x$, respectively.

Secondly, let $\mathbf{C}_{\upsilon_m}$ be the cluster in $\boldsymbol{\Delta}$ which defines **the vicinity of the neighbouring object** $\mathbf{x}_m$. $\mathbf{C}_{\upsilon_m}$ is defined as that subcluster of $\mathbf{C}_i$ which contains, at least, N_{MIN} objects, including $\mathbf{x}_m$:

$$\mathbf{C}_{\upsilon_m} : \upsilon_m = \arg\min_l \{N_l\}, \forall \mathbf{C}_l \subset \mathbf{C}_i \mid \mathbf{x}_m \in \mathbf{C}_l,\ N_l \geq N_{MIN} \tag{4.3}$$

where N_l is the number of objects in cluster $\mathbf{C}_l$. Since the size of the local vicinity is relative with respect to the total size of the dataset, the value of N_{MIN} is defined as 1% of the number of clusters in $\mathbf{X}$ ($N_{MIN} = 0.01\,N$).

Thus, the LSS criterion determines that $\mathbf{C}_i$ is suitable for merging with $\mathbf{C}_j$ **if the standard score statistic of** $\mathbf{C}_{\upsilon_m}$ (ss_{υ_m}) **is greater than or equal to the following dynamic merging threshold**:

$$\begin{aligned} &lss_{th}(d_{ij}, \mathbf{C}_{\upsilon_m}, \mathbf{C}_i, N_{MIN}) = \\ &lss_{th}(d_{ij}, d_{\upsilon_m}, \mu_{\upsilon_m}, \sigma_{\upsilon_m}, n_{\gamma\upsilon_m}, cd_{\upsilon_m}, N_{\upsilon_m}, cd_i, N_i, N_{MIN}) = \\ &\qquad \left(\frac{\gamma'_{\upsilon_m} - \mu'_{\upsilon_m}}{\sqrt{\sigma'_{\upsilon_m} - \mu'^2_{\upsilon_m}}}\right) \Phi(cd_{\upsilon_m}, N_{\upsilon_m}, cd_i, N_i)\ \Psi_L(N_i, N_{MIN}) \end{aligned} \tag{4.4}$$

where γ'_{υ_m} is the gap that would appear if $\mathbf{C}_{\upsilon_m}$ merged with $\mathbf{C}_j$ $\left(\gamma'_{\upsilon_m} = d_{ij} - d_{\upsilon_m}\right)$. Similarly, μ'_{υ_m} and σ'_{υ_m} are the first and second moments of $\overline{\gamma}'_{\upsilon_m}$, which is the sample consisting of the union of $\overline{\gamma}_{\upsilon_m}$ and γ'_{υ_m} $\left(\overline{\gamma}'_{\upsilon_m} = \overline{\gamma}_{\upsilon_m} \cup \gamma'_{\upsilon_m}\right)$, respectively:

$$\mu'_{\upsilon_m} = \frac{\mu_{\upsilon_m} n_{\gamma\upsilon_m} + \gamma'_{\upsilon_m}}{n_{\gamma\upsilon_m} + 1} \quad , \quad \sigma'_{\upsilon_m} = \frac{\sigma_{\upsilon_m} n_{\gamma\upsilon_m} + \gamma'^2_{\upsilon_m}}{n_{\gamma\upsilon_m} + 1} \tag{4.5}$$

Therefore, the **merging rule** derived from the LSS criterion is defined as follows:

- If the LSS criterion is met from the vicinities of $\mathbf{x}_m$ $(ss_{\upsilon_m} \geq lss_{th}(d_{ij}, \mathbf{C}_{\upsilon_m}, \mathbf{C}_i, N_{MIN}))$ and $\mathbf{x}_p$ $\left(ss_{\upsilon_p} \geq lss_{th}\left(d_{ij}, \mathbf{C}_{\upsilon_p}, \mathbf{C}_j, N_{MIN}\right)\right)$ simultaneously, $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster.
- Otherwise, $\mathbf{C}_i$ and $\mathbf{C}_j$ remain separated and pairwise proximities between objects belonging to $\mathbf{C}_{\upsilon_m}$ and $\mathbf{C}_{\upsilon_p}$ are dismissed from the proximity matrix of the dataset and, hence, not anymore considered for the rest of the agglomeration process.

It is worth noting that the LSS criterion essentially defines a comparison between the largest gap within $\mathbf{C}_{\upsilon_m}$ (Γ_{υ_m}) and the hypothetical gap $\left(\gamma'_{\upsilon_m}\right)$ that would appear on $\mathbf{C}_{\upsilon_m}$ as a result of its hypothetical merging with $\mathbf{C}_j$ (see the green dashed link in Figure 4.3b); if γ'_{υ_m} is not significantly higher than Γ_{υ_m}, $\mathbf{C}_i$ will be suitable for merging with $\mathbf{C}_j$.

Finally, on the one hand, $\Phi(cd_{\upsilon_m}, N_{\upsilon_m}, cd_i, N_i)$ is a **density factor** whose purpose is to relax the LSS criterion when the neighbouring object is located in a low-density region of its cluster:

$$\Phi(cd_{\upsilon_m}, N_{\upsilon_m}, cd_i, N_i) = \Phi(dr_i^{\upsilon_m}) = \frac{0.75}{1 + e^{10\left(dr_i^{\upsilon_m} - 0.8\right)}} + 0.25 \tag{4.6}$$

being $dr_i^{\upsilon_m}$ the *density rate*, which indicates the density in the vicinity of the neighbouring object $(\mathbf{C}_{\upsilon_m})$ with respect to the density of the candidate cluster $(\mathbf{C}_i)$:

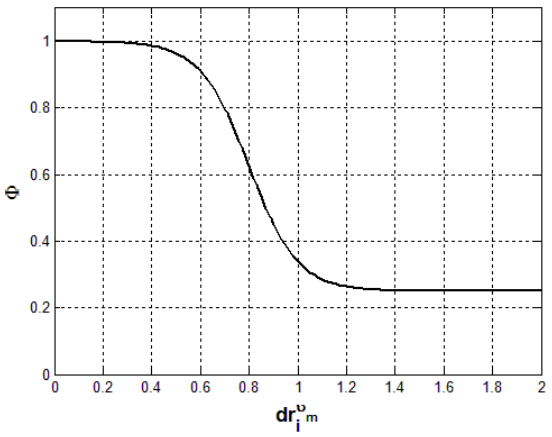
$$dr_i^{\upsilon_m} = \left(\frac{cd_{\upsilon_m}}{N_{\upsilon_m}}\right)\left(\frac{N_i}{cd_i}\right) \tag{4.7}$$

The factor $\left(\frac{cd_x}{N_x}\right)$ is an estimation of the average proximity between objects in cluster $\mathbf{C}_x$, so that it provides an estimated measure of the density of $\mathbf{C}_x$, where N_x is the number of objects in $\mathbf{C}_x$ and cd_x is the cumulative proximity level in $\mathbf{C}_x$ (*i.e.* the sum of the proximity levels of the clusters nested in $\mathbf{C}_x$ and the proximity level of $\mathbf{C}_x$):

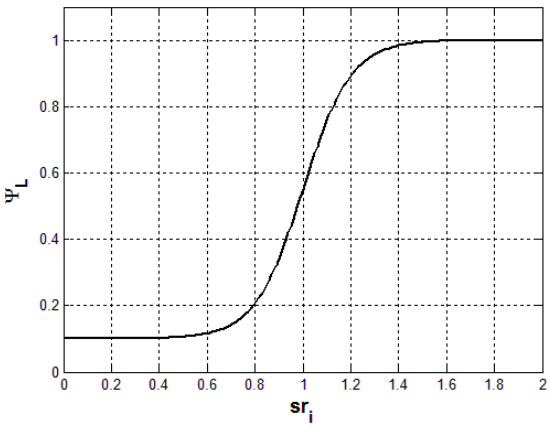
$$cd_x = \sum_{l=1}^{x} d_l, \forall\, \mathbf{C}_l \subseteq \mathbf{C}_x \tag{4.8}$$

Thus, the *density rate* defined in equation 4.7 ($dr_i^{\upsilon_m}$), which results from the ratio between the estimated densities of $\mathbf{C}_{\upsilon_m}$ and $\mathbf{C}_i$, gives a measure of how dense is $\mathbf{C}_{\upsilon_m}$ with respect to $\mathbf{C}_i$: if $\mathbf{C}_{\upsilon_m}$ is located in a high-density region of $\mathbf{C}_i$, the value of $dr_i^{\upsilon_m}$ will tend to be lower than 1 ($dr_i^{\upsilon_m} < 1$); if $\mathbf{C}_{\upsilon_m}$ is located in a low-density region of $\mathbf{C}_i$, the value of $dr_i^{\upsilon_m}$ will tend to be similar to or higher than 1 ($dr_i^{\upsilon_m} \gtrsim 1$).

Hence, as shown in Figure 4.4a, in case that the neighbouring object is located in a high-density region, the density factor defined in equation 4.6 (Φ) keeps approximately equal to 1, which has no effect; however, if the neighbouring object is located in a low-density region, the value of Φ diminishes, which causes a decreasing of the value of lss_{th} and the LSS criterion is therefore relaxed.



(a) Detail of the density factor $\Phi(dr_i^{\upsilon_m})$.



(b) Detail of the size factor $\Psi_L(sr_i)$.

Figure 4.4: LSS criterion: dynamic factors present in the merging threshold lss_{th}.

And on the other hand, $\Psi_L(N_i, N_{MIN})$ is a **size factor** whose purpose (similar to the purpose of the factors $widen_{fact}$ and $delta_{fact}$ defined by Fred and Leitão (2003); see equation 3.24) is to inhibit the LSS criterion when the size of the candidate cluster is small and, hence, to avoid local rejections at too early stages of the agglomeration process:

$$\Psi_L(N_i, N_{MIN}) = \Psi_L(sr_i) = \frac{0.9}{1 + e^{-10(sr_i - 1)}} + 0.1 \tag{4.9}$$

being sr_i the *size rate*, which indicates the size of the candidate cluster ($\mathbf{C}_i$) with respect to the total size of the dataset:

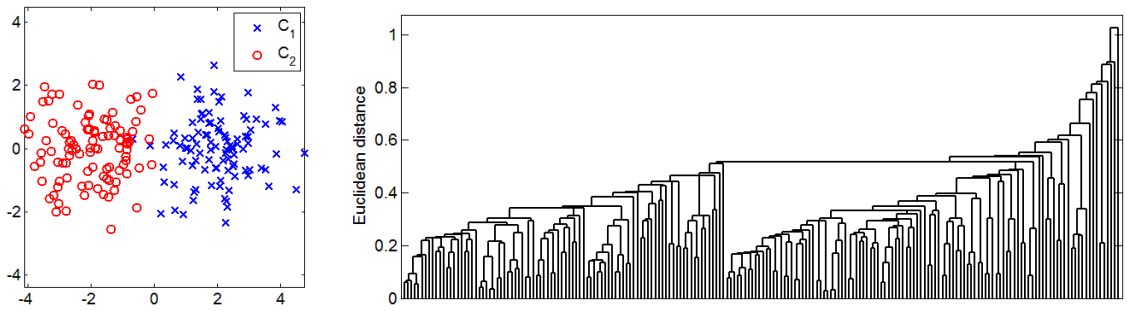
$$sr_i = \frac{N_i}{10\, N_{MIN}} \tag{4.10}$$

Thus, if the size of $\mathbf{C}_i$ is less than $10\,N_{MIN}$ (10% of the total size of the dataset), the value of sr_i is less than 1, whereas sr_i is greater than 1 otherwise. Therefore, as shown in Figure 4.4b, the inhibition of the LSS criterion performed by the size factor Ψ_L disappears as the size of the candidate cluster exceeds the 10% of the total size of the dataset.

4.1.2 The GCSS cluster merging criterion

Whilst it does go beyond the cluster isolation criterion proposed by Fred and Leitão (2000), the LSS criterion may certainly be insufficient by itself in case of dealing with overlapped clusters –or with non-overlapped but highly close clusters–, whose merging can easily give rise to gaps not large enough to cause a rejection, not even in local terms. Hence, the parameter-free AHC algorithm proposed in the next section of the present chapter makes also use of a global cluster merging criterion able to avoid the merging of both touching and overlapped clusters.

To that effect, let the *2Gauss* dataset shown in Figure 4.5a be considered (see section 5.2.2.2 for further details). It is a 2-dimensional synthetic dataset consisting of two partially overlapped Gaussian clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) of the same size ($N_1 = N_2 = 100$) and variance.



(a) *2Gauss* dataset: scatterplot. ***(b)*** *2Gauss* dataset: SL-dendrogram.

Figure 4.5: The *2Gauss* dataset.

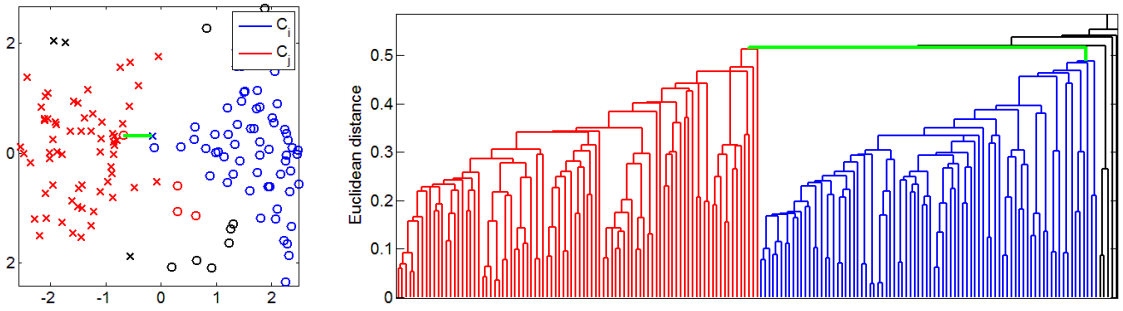
Whereas Fred and Leitão (2003, Section 5.2, pages 951–952) use this kind of dataset (mixture of Gaussian clusters) in order to explore some performance limits of their first version of the SL-DID algorithm, the *2Gauss* dataset allows to gain an insight into how partially overlapped clusters can be identified from an AHC algorithm based on the SL method. Thus, the SL-dendrogram on the *2Gauss* dataset is shown in Figure 4.5b.

At a certain stage of the agglomeration process, a pair of clusters candidate for merging ($\mathbf{C}_i$ and $\mathbf{C}_j$) arises. Being two approximately balanced clusters ($N_i = 84\,, N_j = 90$), they both merge into a new cluster in the dendrogram ($\mathbf{C}_k$) as shown in Figure 4.6. This precise merging presents several peculiarities:

- Due to the partial overlapping of the two Gaussian clusters that constitute the dataset, both candidate clusters include elements belonging to the two ground-truth clusters. Nonethe-

less, $\mathbf{C}_i$ and $\mathbf{C}_j$ are mostly composed by objects belonging to $\mathbf{C}_1$ and $\mathbf{C}_2$, respectively. Thus, **it is a troublesome merging**, since, in case of not being rejected, it makes impossible to distinguish between $\mathbf{C}_1$ and $\mathbf{C}_2$ in the final clustering solution. Therefore, **it should be rejected**.

- The merging does not cause any significantly high gap with respect to the gaps already existing within $\mathbf{C}_i$ or $\mathbf{C}_j$ (see the green-coloured link in Figure 4.6b).
- Beyond this particular example, the neighbouring objects ($\mathbf{x}_m \in \mathbf{C}_i$ and $\mathbf{x}_p \in \mathbf{C}_j$) that give rise to a merging between two partially overlapped clusters can easily belong as much to a low density region as to a high density region in their respective clusters.



(a) Candidate clusters $\mathbf{C}_i$ and $\mathbf{C}_j$.

(b) $\mathbf{C}_i$ and $\mathbf{C}_j$ in the SL-dendrogram.

Figure 4.6: *2Gauss* dataset: detail of the agglomeration process. ***(a)*** Zoom in on the scatterplot in Figure 4.5a. Candidate clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ are depicted by colours (see legend); the proximity between neighbouring objects is green-coloured. ***(b)*** Zoom in on the dendrogram in Figure 4.5b. $\mathbf{C}_i$ and $\mathbf{C}_j$ can be identified according to their respective colours in Figure 4.2a; the link resulting from the merging is green-coloured.

As a consequence, **mergings of this kind can easily not cause any gap to be high enough** to fall not only in the tail of the DIDs of $\mathbf{C}_i$ and $\mathbf{C}_j$, but even in the tail of the DIDs existing in the vicinities of the neighbouring objects. Thus, such mergings will hardly be rejected, neither locally nor globally, by means of merging criteria or statistical rules derived only from the proximity level of the hypothetical new cluster. Therefore, again, a question arises:

- How touching and overlapped clusters can be identified in the context of a SL-dendrogram without involving a significant increase of the computational cost of the clustering process?

The answer to that question is provided in the present thesis by making use of one of the parameters the density factor Φ utilised in the LSS criterion (see equation 4.6) is based on: **the cumulative proximity level** of a cluster, which is defined in equation 4.8 as the sum of its proximity level and the proximity levels of its nested clusters in the dendrogram.

As shown in Figure 4.7, the cumulative proximity level of clusters in the SL-dendrogram of the *2Gauss* dataset shown in Figure 4.6b presents an abrupt increase when clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into $\mathbf{C}_k$ (see the green mark in Figure 4.7), which causes a sudden interruption of the progressive increment of the cumulative proximity levels throughout the agglomeration process.

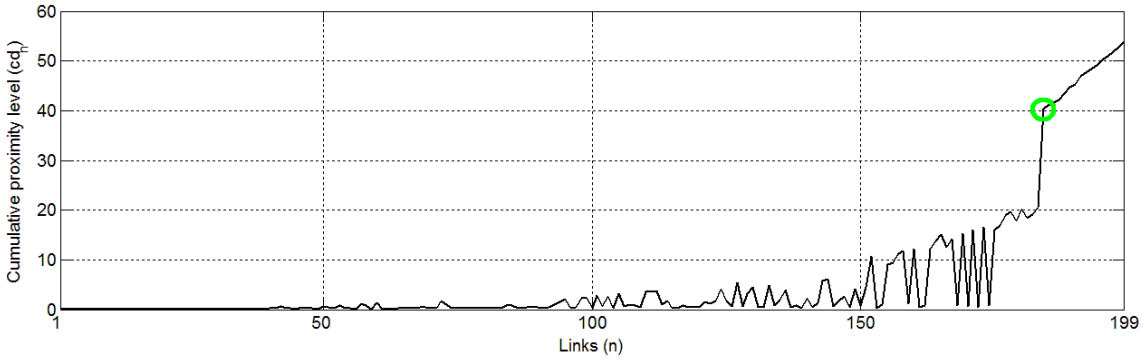


Figure 4.7: Cumulative proximity levels in the SL-dendrogram of the *2Gauss* dataset. The green mark indicates the cumulative proximity level of cluster $\mathbf{C}_k$ (cd_k).

It is worth noting that the abrupt increase cd_k involves in comparison with the cumulative proximities of the rest of clusters in the dendrogram is a differential characteristic that may allow to identify undesired mergings of touching or overlapped clusters and, therefore, to reject them.

Thus, let $\mathbf{X}$ be a dataset constituted by a set of N D-dimensional objects. Given an agglomeration process of the objects in $\mathbf{X}$ based on the SL method, let $\mathbf{C}_i$ and $\mathbf{C}_j$ be a pair of clusters of N_i and N_j objects, respectively, candidate for merging into a new cluster $\mathbf{C}_k$ and let d_{ij} be the proximity between them. The **GCSS criterion** determines whether $\mathbf{C}_i$ and $\mathbf{C}_j$ should merge into a new cluster or not and it is defined as follows.

Firstly, let $\mathbf{C}_x$ be any given cluster in the dendrogram $\mathbf{\Delta}$ resulting from the agglomeration process of the objects in $\mathbf{X}$ and let $\overline{cd}_x$ be the sample consisting of its own cumulative proximity level (cd_x) and the cumulative proximity levels of its nested clusters in $\mathbf{\Delta}$. **The cumulative standard score statistic of cluster** $\mathbf{C}_x$ (css_x) is defined as the standard score of cd_x with respect to $\overline{d}_x$:

$$css_x = \frac{cd_x - c\mu_x}{\sqrt{c\sigma_x - c\mu_x^2}} \tag{4.11}$$

where $c\mu_x$ and $c\sigma_x$ are the first and second moments of $\overline{cd}_x$, respectively:

$$\mu_x = \frac{1}{n_{dx}} \sum_{l=1}^{n_{dx}} \overline{cd}_{x_l} \quad , \quad \sigma_x = \frac{1}{n_{dx}} \sum_{l=1}^{n_{dx}} \overline{cd}_{x_l}^2 \tag{4.12}$$

being $\overline{cd}_{x_l}$ the lth observation in $\overline{cd}_x$ and n_{dx} the length of $\overline{cd}_x$ (*i.e.* the number of non-singleton clusters nested within $\mathbf{C}_x$). Continuing with the example of the *2Gauss* dataset, the values of the css statistic for the clusters in the *2Gauss* dataset SL-dendrogram are shown in Figure 4.8. It is worth noting that the maximum value is reached for the critical cluster $\mathbf{C}_k$ (see the green-coloured link in Figure 4.6b and the green mark in Figure 4.8). Thus, the abrupt increase of the cd parameter illustrated in Figure 4.7 (cd_k) leads into a maximum value of the css statistic (css_k), which is significantly higher than the rest of the previous values of the statistic.

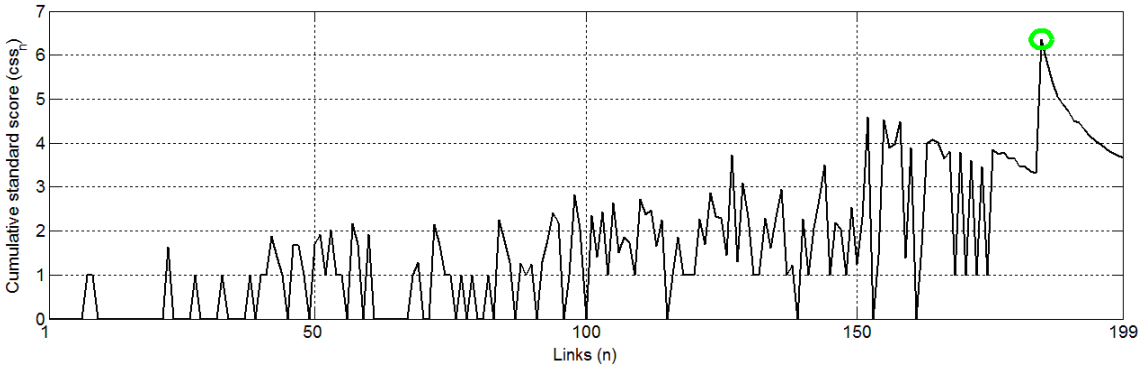


Figure 4.8: Cumulative standard score of the clusters in the SL-dendrogram of the *2Gauss* dataset. The green mark indicates the cumulative standard score of cluster $\mathbf{C}_k$ (css_k).

In this way, the GCSS criterion determines that the union between $\mathbf{C}_i$ and $\mathbf{C}_j$ into a new cluster $\mathbf{C}_k$ is a suitable merging **if their cumulative standard score statistics** $(css_i$ **and** $css_j)$ **are greater than or equal to the following dynamic merging threshold**:

$$\begin{aligned} &gcss_{th}(\mathbf{C}_k, \mathbf{C}_i, \mathbf{C}_j, N_{MIN}) = \\ &gcss_{th}(css_k, N_i, \gamma_i, \mu_i, \sigma_i, N_j, \gamma_j, \mu_j, \sigma_j, N_{MIN}) = \\ &\qquad css_k \, \Upsilon(N_i, N_j) \; \Psi_G(N_i, \gamma_i, \mu_i, \sigma_i, N_j, \gamma_j, \mu_j, \sigma_j, N_{MIN}) \end{aligned} \tag{4.13}$$

where css_k is the cumulative standard score of $\mathbf{C}_k$, $N_{MIN} = 0.01\,N$, $\gamma_i = d_{ij} - d_i$, $\gamma_j = d_{ij} - d_j$ and μ_i, σ_i, μ_j and σ_j are defined according to equation 4.2.

Therefore, the **merging rule** derived from the GCSS criterion is defined as follows:

- If the GCSS criterion is simultaneously met from both $\mathbf{C}_i$ $(css_i \geq gcss_{th}(\mathbf{C}_k, \mathbf{C}_i, \mathbf{C}_j, N_{MIN}))$ and $\mathbf{C}_j$ $(css_j \geq gcss_{th}(\mathbf{C}_k, \mathbf{C}_i, \mathbf{C}_j, N_{MIN}))$, $\mathbf{C}_i$ and $\mathbf{C}_j$ merge into a new cluster.
- Otherwise, the merging between $\mathbf{C}_i$ and $\mathbf{C}_j$ is rejected in global terms, so that they remain separated and all the pairwise proximities between objects belonging to $\mathbf{C}_i$ and $\mathbf{C}_j$ are dismissed from the proximity matrix of the dataset in case of rejection and, hence, not anymore considered for the rest of the agglomeration process.

Similarly to the merging rule derived from the LSS criterion, neither of the candidate clusters is isolated in case of global rejection, so that flexibility is maximised; *i.e.* the possibility that both $\mathbf{C}_i$ and $\mathbf{C}_j$ merge with other clusters in further stages of the agglomeration process remains unaltered.

It is worth noting that the GCSS criterion essentially compares the cumulative proximity level of the hypothetical new cluster (cd_k) with the cumulative proximity levels of both candidate clusters $(cd_i$ and $cd_j)$. These cumulative proximity levels are compared into the context of the distributions of cumulative proximity levels present in their respective clusters, modelled by means of css_k, css_i and css_j, respectively. Hence, in essence, if cd_k involves an increment in the context of $\mathbf{C}_k$ higher than both the increments cd_i and cd_j involve in the contexts of $\mathbf{C}_i$ and $\mathbf{C}_j$, respectively (*i.e.* if css_k is higher than both css_i and css_j), $\mathbf{C}_i$ and $\mathbf{C}_j$ will not be suitable for merging in global terms.

Finally, on the one hand, $\Upsilon(N_i, N_j)$ is a **balance factor** whose purpose is to relax the GCSS criterion when the candidate clusters $\mathbf{C}_i$ and $\mathbf{C}_j$ are far from being balanced each other:

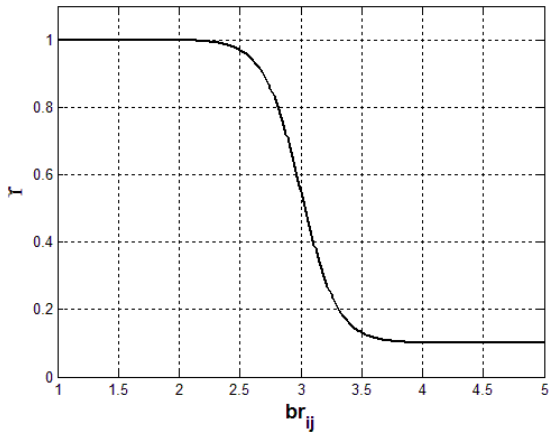
$$\Upsilon(N_i, N_j) = \Upsilon(br_{ij}) = \frac{0.9}{1 + e^{20\left(\frac{br_{ij}}{3} - 1\right)}} + 0.1 \tag{4.14}$$

being br_{ij} the *balance rate*, which indicates how unbalanced $\mathbf{C}_i$ and $\mathbf{C}_j$ are:

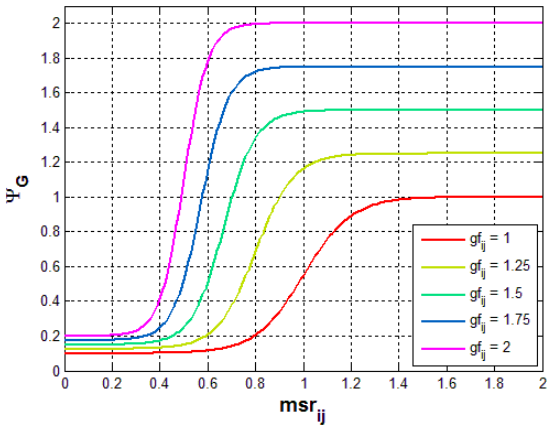
$$br_{ij} = \frac{\max\{Ni, Nj\}}{\min\{Ni, Nj\}} \tag{4.15}$$

Thus, br_{ij} is equal to the size of the most populated candidate cluster divided by the size of the less populated candidate cluster. Therefore, being a ratio always greater than or equal to 1 ($br_{ij} \geq 1$), the value of br_{ij} is close to 1 to the extent that $\mathbf{C}_i$ and $\mathbf{C}_j$ are balanced ($br_{ij} \approx 1 \Leftrightarrow N_i \approx N_j$) and it reaches its minimum when $\mathbf{C}_i$ and $\mathbf{C}_j$ are completely balanced ($br_{ij} = 1 \Leftrightarrow N_i = N_j$)

Hence, as shown in Figure 4.9a, in case that candidate clusters are approximately balanced, the balance factor defined in equation 4.14 keeps approximately equal to 1, which has no effect; however, if candidate clusters are clearly unbalanced, the value of Υ diminishes, which causes a decreasing of the value of $gcss_{th}$ and the GCSS criterion is therefore relaxed. Since its purpose is to provide the ability of identifying approximately balanced and partially overlapped clusters (*e.g.* see Figure 4.5), this behaviour is worthwhile in the GCSS criterion with the aim of avoiding undesired rejections of mergings between unbalanced touching clusters, which are particularly likely when dealing with clusters of uniformly distributed objects (see section 5.2.1 for further details).



(a) Detail of the balance factor $\Upsilon(br_{ij})$.



(b) Detail of the size factor $\Psi_G(msr_i, gf_{ij})$.

Figure 4.9: GCSS criterion: dynamic factors present in the merging threshold $gcss_{th}$.

And on the other hand, $\Psi_G(N_i, \gamma_i, \mu_i, \sigma_i, N_j, \gamma_j, \mu_j, \sigma_j, N_{MIN})$ is a **size factor** whose purpose is to inhibit the GCSS criterion when the size of candidate clusters is not large enough and, hence, to avoid global rejections at too early stages of the agglomeration process:

$$\Psi_G(N_i, \gamma_i, \mu_i, \sigma_i, N_j, \gamma_j, \mu_j, \sigma_j, N_{MIN}) = \Psi(msr_{ij}, gf_{ij}) = gf_{ij} \frac{0.9}{1 + e^{-10((msr_{ij}\, gf_{ij}) - 1)}} + 0.1 \tag{4.16}$$

being msr_{ij} the *maximum size rate*, which indicates the size of the largest candidate cluster with respect to the total size of the dataset:

$$msr_{ij} = \frac{\max\{N_i, N_j\}}{20\, N_{MIN}} \tag{4.17}$$

and gf_{ij} the *gap factor*, whose purpose is to disable the size factor $\Psi(msr_{ij}, gf_{ij})$ to the extent that, regardless of the size of $\mathbf{C}_i$ and $\mathbf{C}_j$, the merging causes large gaps on the candidate clusters:

$$gf_{ij} = \max\left\{1, \frac{\gamma_i - \mu_i}{\sqrt{\sigma_i - \mu_i^2}}, \frac{\gamma_j - \mu_j}{\sqrt{\sigma_j - \mu_j^2}}\right\} \tag{4.18}$$

Thus, if the size of the largest candidate cluster is less than $20\,N_{MIN}$ (20% of the total size of the dataset), the value of msr_{ij} is less than 1, whereas msr_{ij} is greater than 1 otherwise. In addition, if γ_i and/or γ_j lie more than a standard deviation above the mean value of the distribution of gaps within its respective cluster, the value of gf_{ij} is greater than 1, whereas gf_{ij} is equal to 1 otherwise. Therefore, as shown in Figure 4.9b, the inhibition of the GCSS criterion performed by the size factor $\Psi_G\,(msr_{ij}, gf_{ij} = 1)$ disappears as the size of the largest candidate cluster exceeds the 20% of the total size of the dataset. Moreover, inasmuch as gf_{ij} increases its value, this inhibition tends to disappear by means of both a decrease of the threshold value of msr_{ij} (smaller candidate clusters lead to higher values of Ψ_G) and an increase of the top value of Ψ_G (by setting $\max\{\Psi_G\} = gf_{ij}$, the GCSS criterion is toughened when $gf_{ij} > 1$).

4.2 The LSS-GCSS algorithm

The cluster merging criteria presented in the previous section give rise to the main contribution of the present thesis: **The LSS-GCSS algorithm** (LSS-GCSS refers to Local Standard Score - Global Cumulative Standard Score), a novel parameter-free AHC algorithm based on both LSS and GCSS criteria and whose flowchart is show in Figure 4.10. Furthermore, a detailed description of LSS-GCSS is provided in Algorithm 5, along with the procedures shown in Algorithms 6, 7 and 8).

This new algorithm presents several noticeable characteristics, specially in comparison with both versions of the SL-DID algorithm (see sections 3.2.1 and 3.2.2), which are next summarised:

- Being an evolution of the SL-DID algorithm, LSS-GCSS is a parameter-free AHC algorithm that combines elements from both graph-based (it also implements the SL method) and model-based (it is also defined by merging criteria that characterise clusters in probabilistic terms) approaches to clustering. In addition, the behaviour of the LSS criterion depends on the density of the regions the neighbouring objects are located, so that the LSS-GCSS algorithm takes some notions belonging to the density-based approach to clustering as well.

- Compared to both versions of SL-DID, LSS-GCSS is a more flexible algorithm, since, firstly, it never isolates any candidate cluster in case of rejection and, secondly, the decoupling of candidate clusters performed in case of local rejection only considers the pairwise proximities belonging to the vicinities of the neighbouring objects (see lines 10–15 in Algorithm 6).

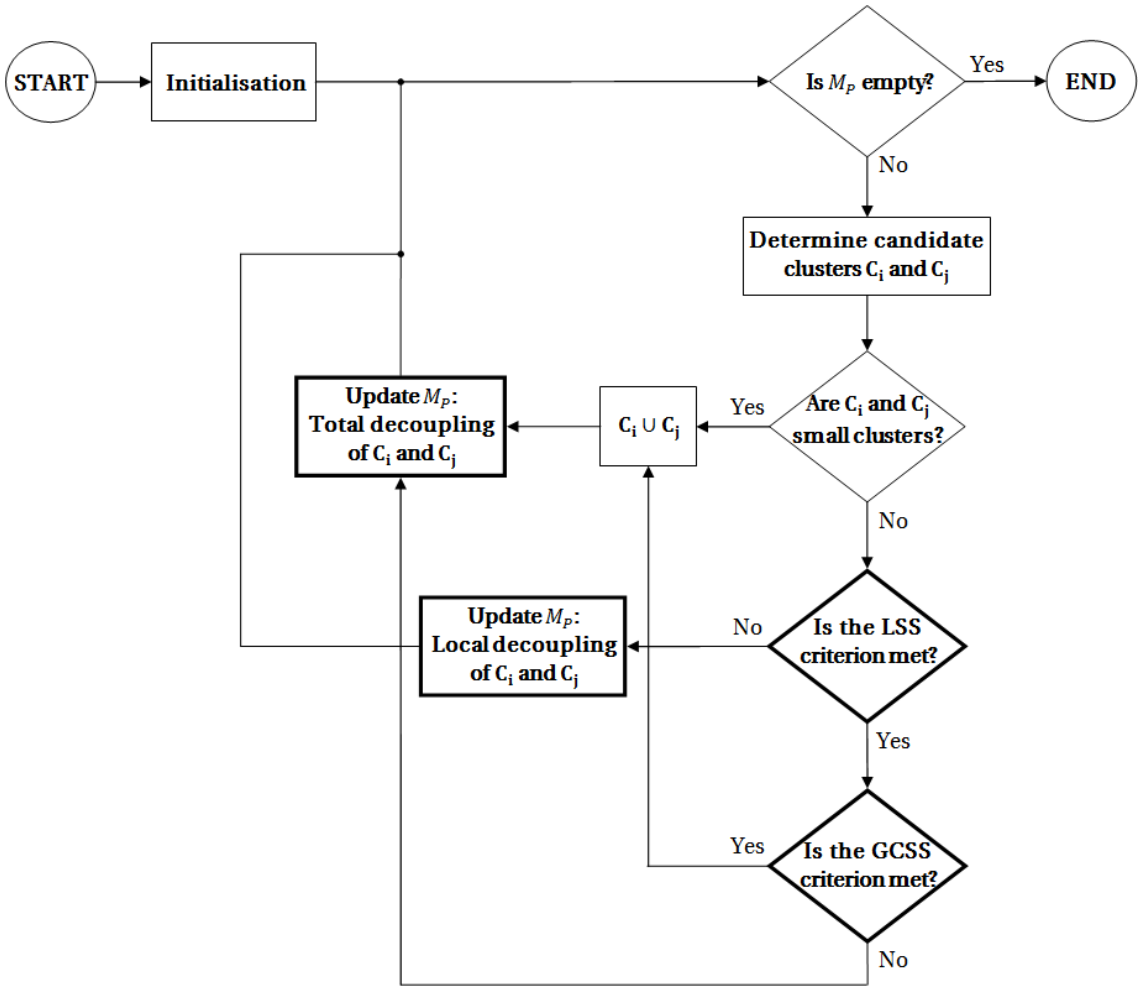


Figure 4.10: Flowchart diagram of LSS-GCSS algorithm.

- Similarly to the second version of SL-DID (see Algorithm 4 in section 3.2.2), LSS-GCSS gives a differential treatment to small candidate clusters. As shown in lines 7 and 9 of Algorithms 6 and 7, respectively, candidate clusters whose size is lower than N_{MIN} are not required to account for the merging criteria. In fact, the merging between $\mathbf{C}_i$ and $\mathbf{C}_j$ always take place in case both candidate clusters include less than N_{MIN} objects. Regarding the cluster size threshold, it is worth noting the difference between N_{LOCAL} and N_{MIN} (see line 7 in Algorithm 5); being both values referred to the size of clusters, N_{LOCAL} is a real value used in the calculation of the dynamic merging thresholds (see line 8 in Algorithm 6 and line 10 in Algorithm 7), whereas N_{MIN} is an integer threshold value used when a direct comparison with a cluster size is required (see lines 7, 10 and 13 in Algorithm 6, line 9 in Algorithm 7 and line 18 in Algorithm 8). Finally, the minimum value of N_{MIN} is limited to 4 (see line 7 in Algorithm 5) inasmuch as the minimum amount of objects required to calculate the first and second moments of a sample of gaps is 4 (4 objects give rise to 2 gaps).

- When dealing with non-small clusters (those whose size is greater than N_{MIN}), both LSS and GCSS criteria are tested in order to accept or reject the merging, which, as shown in Figure 4.10, takes place if and if only both merging criteria are met.

- The dynamic merging thresholds are computed according to equations 4.4–4.7, 4.9 and 4.10 in the case of LSS criterion (see line 8 in Algorithm 6) and according to equations 4.13–4.18 in the case of GCSS criterion (see and line 10 in Algorithm 7).

- Finally, part of the parameters that characterise the hypothetical new cluster ($\mathbf{C}_k$) are calculated as the GCSS criterion is tested (see lines 2–6 in Algorithm 7. The characterisation of $\mathbf{C}_k$ is completed when candidate clusters eventually merge (see lines 4–24 in Algorithm 8). Singleton clusters are excluded from this characterisation (see line 12 in Algorithm 8), since their gaps are in reality just proximities between a pair of neighbouring objects. Moreover, the vicinity of objects is also determined at this point, in the event of ($\mathbf{C}_k$) being non-small and, at least, one of the candidate clusters being small (see lines 18–20 in Algorithm 8).

Algorithm 5 LSS-GCSS algorithm.

1: *Input*: Dataset $\mathbf{X}$
2: **procedure**
3: $M_P : M_P(i,j) \leftarrow d_{\mathbf{x}_i\mathbf{x}_j}, \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X} \mid i \neq j$
4: $v_i \leftarrow 0, \forall i \in \{1, N\}$
5: $\boldsymbol{\lambda} : \lambda_i \leftarrow i, \forall i \in \{1, N\}$ ▷ Initially, $\mathbf{P} = \{\mathbf{C}_1, ..., \mathbf{C}_N\} \mid \mathbf{C}_i = \{\mathbf{x}_i\}$
6: $\boldsymbol{\Delta} : \overline{\Delta}_i \leftarrow [0\ 0\ 0], \forall i \in \{1, N-1\}$
7: $N_{LOCAL} \leftarrow 0.01\,N, \quad N_{MIN} \leftarrow \min\{4, \lfloor N_{LOCAL} \rceil\}$
8: $N_i \leftarrow 1,\ d_i \leftarrow 0,\ n_{\gamma i} \leftarrow 0,\ \Gamma_i \leftarrow 0,\ \mu_i \leftarrow 0,\ \sigma_i \leftarrow 0,\ ss_i \leftarrow 0,\ cd_i \leftarrow 0,\ n_{di} \leftarrow 0,$
$cd_i \leftarrow 0,\ n_{di} \leftarrow 0,\ c\mu_i \leftarrow 0,\ c\sigma_i \leftarrow 0,\ css_i \leftarrow 0,\ \forall i \in \{1, N\}$
9: $k \leftarrow N+1$
10: $\rho \leftarrow false$
11: **while** $M_P \neq \emptyset$ **do**
12: $(m, p) \leftarrow \arg\min_{(q,r)} \{M_P(q,r)\},\ d_{ij} \leftarrow \min\{M_P\}$
13: $i \leftarrow \lambda_m,\ j \leftarrow \lambda_p$
14: LSS ▷ Evaluation of LSS criterion
15: **if** $\neg\rho$ **then**
16: GCSS ▷ Evaluation of GCSS criterion
17: **end if**
18: Update M_P: Dismiss the proximities between objects belonging to $\mathbf{C}_i$ and $\mathbf{C}_j$
19: **if** $\neg\rho$ **then**
20: MERGINGCANDIDATES ▷ A new cluster is created
21: **else**
22: $\rho \leftarrow false$
23: **end if**
24: **end while**
25: **end procedure**
26: *Output*: Label vector $\boldsymbol{\lambda}$ and dendrogram $\boldsymbol{\Delta}$

Algorithm 6 LSS procedure in LSS-GCSS algorithm.

procedure
  $objects \leftarrow [m\ p]$
  $candidates \leftarrow [i\ j]$
  for $l \leftarrow 1, 2$ **do**
    $o \leftarrow objects\,(l)$
    $c \leftarrow candidates\,(l)$
    if $(N_c \geq N_{MIN})$ **then**
      if $(ss_{v_o} < lss_{th}(d_{ij}, d_{v_o}, \mu_{v_o}, \sigma_{v_o}, n_{\gamma v_o}, cd_{v_o}, N_{v_o}, cd_c, N_c, N_{LOCAL}))$ **then**
        $\rho \leftarrow true$ ▷ Local rejection
        if $N_i \geq N_{MIN}$ **then**
          $i \leftarrow v_m$
        end if
        if $N_j \geq N_{MIN}$ **then**
          $j \leftarrow v_p$
        end if
        break for ▷ Break out of the loop
      end if
    end if
  end for
end procedure

Algorithm 7 GCSS procedure in LSS-GCSS algorithm.

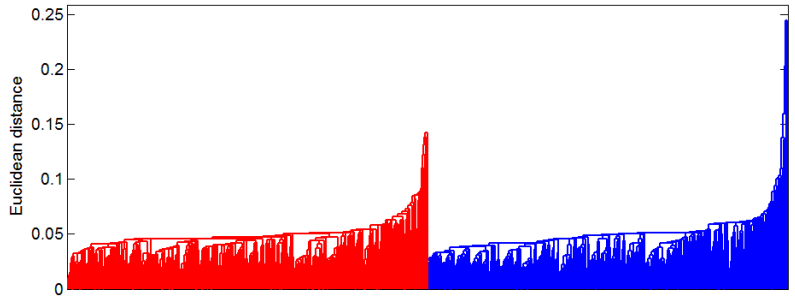
procedure
  $cd_k \leftarrow cd_i + cd_j + d_{ij}$
  $n_{dk} \leftarrow n_{di} + n_{dj} + 1$
  $c\mu_k \leftarrow \frac{c\mu_i\, n_{di} + c\mu_j\, n_{dj} + cd_k}{n_{dk}}$
  $c\sigma_k \leftarrow \frac{c\sigma_i\, n_{di} + c\sigma_j\, n_{dj} + cd_k^2}{n_{dk}}$
  $css_k \leftarrow \frac{cd_k - c\mu_k}{\sqrt{c\sigma_k - c\mu_k^2}}$
  $\gamma_i \leftarrow d_{ij} - d_i$
  $\gamma_j \leftarrow d_{ij} - d_j$
  if $(N_j \geq N_{MIN}) \wedge (N_j \geq N_{MIN})$ **then**
    $g_{th} \leftarrow gcss_{th}(css_k, N_i, \gamma_i, \mu_i, \sigma_i, N_j, \gamma_j, \mu_j, \sigma_j, N_{LOCAL})$
    if $(css_i < g_{th}) \wedge (css_j < g_{th})$ **then**
      $\rho \leftarrow true$ ▷ Global rejection
    end if
  end if
end procedure

Algorithm 8 MERGINGCANDIDATES procedure in LSS-GCSS algorithm.

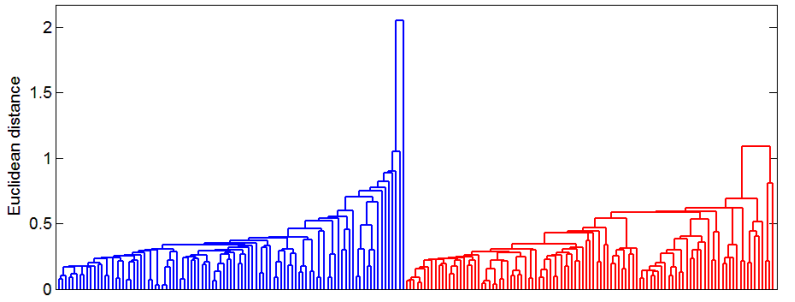
procedure
  $\lambda_l \leftarrow k,\ \forall\, l \in \{1, N\} \mid \lambda_l = i,\ \lambda_l = j$ ▷ $\mathbf{C}_k = \mathbf{C}_i \cup \mathbf{C}_j$
  $\overline{\Delta}_{(k\text{–}N)} \leftarrow [\, i\ j\ d_{ij}]$
  $N_k \leftarrow N_i + N_j$
  $d_k \leftarrow d_{ij}$
  $n_{\gamma k} \leftarrow n_{\gamma i} + n_{\gamma j}$
  $\Gamma_k \leftarrow \max\{\Gamma_i,\, \Gamma_j\}$
  $\mu_k \leftarrow \mu_i\, n_{\gamma i} + \mu_j\, n_{\gamma j}$
  $\sigma_k \leftarrow \sigma_i\, n_{\gamma i} + \sigma_j\, n_{\gamma j}$
  for $l \leftarrow 1, 2$ **do**
    $c \leftarrow candidates\,(l)$
    if $(N_c > 1)$ **then** ▷ Gaps of singleton clusters are not considered
      $n_{\gamma k} \leftarrow n_{\gamma k} + 1$
      $\Gamma_k \leftarrow \max\{\Gamma_k,\, \gamma_c\}$
      $\mu_k \leftarrow \mu_k + \gamma_c$
      $\sigma_k \leftarrow \sigma_k + \gamma_c^2$
    end if
    if $(N_c < N_{MIN}) \wedge (N_k \geq N_{MIN})$ **then**
      $v_l \leftarrow k,\ \forall\, l \in \{1, N\} \mid \lambda_l = c$ ▷ Vicinity of the objects belonging to $\mathbf{C}_c$
    end if
  end for
  $\mu_k \leftarrow \frac{\mu_k}{n_{\gamma k}}$
  $\sigma_k \leftarrow \frac{\sigma_k}{n_{\gamma k}}$
  $ss_k \leftarrow \frac{\Gamma_k - \mu_k}{\sqrt{\sigma_k - \mu_k^2}}$
  $k \leftarrow k+1$
end procedure

Thus, similarly to SL-DID, LSS-GCSS algorithm automatically generates a twofold clustering solution composed of an HPC solution (represented by $\boldsymbol{\lambda}$, which identifies K hard clusters) and an AHC solution (represented by $\boldsymbol{\Delta}$, which includes K dendrograms –one per hard cluster–). The insight into the nature of the data provided by this kind of clustering solution is illustrated in Figure 4.11, which shows the clustering solutions obtained by means of the LSS-GCSS algorithm on *2bars* (see section 4.1.1), *2Gauss* (see section 4.1.2) and *4toy* (see sections 3.1.3 and 3.2.1) datasets.

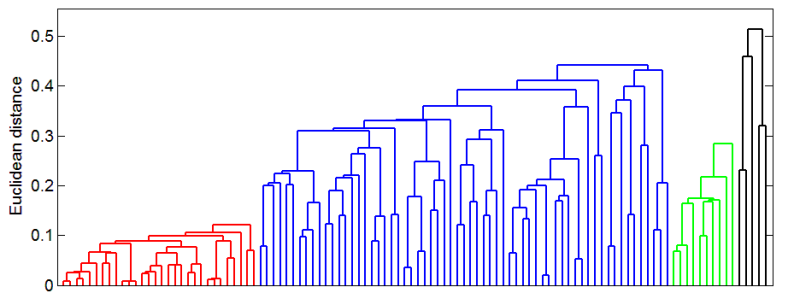
It is worth noting that optimal HPC solutions are reached for *2bars* and *4toy* datasets (ground truth solutions are equalled in both cases), whereas, since it contains partially overlapped clusters, the HPC solution on *2Gauss* dataset is not optimal, but highly close to the ground truth ($CI = 0.97$). Furthermore, $CPCC$ values of the dendrograms included in the AHC solutions are shown in Table



(a) *2bars* dataset: AHC solution. ***(b)*** *2bars* dataset: HPC solution.



(c) *2Gauss* dataset: AHC solution. ***(d)*** *2Gauss* dataset: HPC solution.



(e) *4toy* dataset: AHC solution. ***(f)*** *4toy* dataset: HPC solution.

Figure 4.11: AHC and HPC solutions on *2bars*, *2Gauss* and *4toy* datasets by means of LSS-GCSS algorithm. ***(a) (c) (e)*** Every AHC solution comprises as many dendrograms as clusters are identified in its respective HPC solution. Dendrograms in AHC solutions are validated in Table 4.1 by means of their $CPCC$ values. ***(b) (d) (f)*** HPC solutions on *2bars* and *4toy* dataset match the ground truth 100% ($CI = 1$), whereas HPC solution on *2Gauss* dataset in a quasi-optimal solution ($CI = 0.97$).

2bars		*2Gauss*		*4toy*			
$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$	$\mathbf{C}_4$
0.358	0.372	0.541	0.617	0.672	0.72	0.643	0.729

Table 4.1: $CPCC$ values for dendrograms in Figure 4.11.

4.1. The absence of significantly close to 0 (or even negative) $CPCC$ values validates the obtained AHC solutions and indicates that the resultant dendrograms properly represent the internal structure of clusters in the three datasets (*e.g.* the shape of dendrograms in Figures 4.11a and 4.11c indicates the variable density of objects present in clusters belonging to both *2bars* and *2Gauss* datasets, in contrast with the more uniform distributions suggested by dendrograms in Figure 4.11e).

However, $CPCC$ values in *2bars* dataset are clearly lower in comparison with the rest, which is a noticeable fact regarding that the HPC solution on the *2bars* dataset matches the ground truth $(CI = 1)$; as aforementioned in sections 3.1.3 and 3.2.1, this fact may easily be suggesting that objects in *2bars* dataset are not hierarchically distributed (see section 2.4.4 for further details).

4.3 Computational requirements of LSS-GCSS algorithm

Despite involving the computation of more complex merging criteria than both versions of SL-DID algorithm, the behaviour of LSS-GCSS algorithm concerning its computational requirements remains $O(N^2)$ in storage terms and $O(N^2 \log N)$ in time terms.

Storage requirements of LSS-GCSS algorithm

As aforementioned in sections 2.2, 3.1 and 3.2.1, the computation of M_P, $\boldsymbol{\lambda}$ and $\boldsymbol{\Delta}$ (see lines 3, 5 and 6 in Algorithm 5) requires the storage of $\frac{N^2-N}{2}$, N and $3\,(N-1)$ values, respectively. In addition, each of the remaining variables (see lines 4 and 8 in Algorithm 5, lines 2–6 in Algorithm 7 and lines 4–24 in Algorithm 8) requires the storage of $2N-K$ values, where K is the number of hard clusters the algorithm eventually identifies $(K \in [1, N])$.

Therefore, the most restrictive requirement correspond to the storage of M_P, which leads LSS-GCSS to be an $O(N^2)$ algorithm in storage terms.

Time requirements of LSS-GCSS algorithm

Concerning the computational requirements of LSS-GCSS algorithm in time terms, an important consideration has to be firstly noticed: they cannot be calculated *a priori* in deterministic terms. Let the computation of the LSS-GCSS algorithm on two different datasets ($\mathbf{X}_1$ and $\mathbf{X}_2$) of the same size $(N_1 = N_2 = N)$ be considered. Even in the case that LSS-GCSS identifies the same number of clusters for both datasets $(K_1 = K_2 = K)$, the number of iterations required to complete both clustering processes may easily be different, since it does not depend on either N or K, but on the way the values of M_P are dismissed throughout both agglomeration processes (see line 11 in Algorithm 5).

Furthermore, Figure 4.10 clearly shows that every iteration in the clustering process has four possible endings (merging of small candidate clusters, merging locally rejected, merging globally rejected, or merging of large candidate clusters), each one of which involves both a different computation time and dismissing a different amount of values from M_P. Hence, the total computation time of the LSS-GCSS algorithm cannot be determined with exactitude in advance, since it always depend on the particularities of every dataset.

Thus, the aim of the following analysis is to establish an upper bound for the computational requirements of the LSS-GCSS algorithm in time terms:

- As previously stated in section 3.2.1, the calculation of M_P involves a N^2-dependent computation time. Besides, the step consisting on finding the minimum value of M_P at every iteration of the algorithm according to the SL method (see line 12 in Algorithm 5) can be optimised by means of a prior sorting of the values in M_P. Algorithms like Quicksort, Mergesort or Heapsort can perform this task involving a $\left(N^2 \log N\right)$-dependent computation time.

- As aforementioned, the total amount of iterations required to complete the clustering process is *a priori* unknown. However, it will never overpass the maximum amount of different proximity values present in M_P; *i.e.* it will always be lower than $\frac{N^2-N}{2}$, which is an adequate but poor estimation of an upper bound, since proximities are dismissed one at a time only when singleton clusters merge.

- The LSS procedure (see Algorithm 6) is computed once per every iteration that entails, at least, one large candidate cluster. The cost of this procedure resides in the calculation of the dynamic merging threshold of the LSS criterion (line 8 in Algorithm 6), which is computed according to equations 4.4–4.7, 4.9 and 4.10. These equations comprise a constant (*i.e.* independent from N) number of operations, so that the LSS procedure involves a N-independent computation time per iteration.

- The GCSS procedure (see Algorithm 7) is composed of two different parts.

  On the one hand, lines 2–8 are computed once per every iteration that does not entail a local rejection of the merging. These lines comprise a constant (*i.e.* independent from N) number of operations, so that this part of GCSS procedure involves a N-independent computation time per iteration.

  On the other hand, the second part of the GCSS procedure is computed once per every iteration with two large candidate clusters whose merging has not been locally rejected. Its cost resides in the calculation of the dynamic merging threshold of the GCSS criterion (line 10 in Algorithm 7), which is computed according to equations 4.13–4.18. These equations comprise a constant (*i.e.* independent from N) number of operations, so that this part of GCSS procedure involves an N-independent computation time per iteration as well.

  Hence, the GCSS procedure involves a N-independent total computation time per iteration.

- The update of M_P (line 18 in Algorithm 5) involves dismissing $\frac{N^2-N}{2}$ values from M_P throughout the entire algorithm. Hence, its total cost is $O\left(N^2\right)$, regardless of the number of iterations eventually completed.

- Finally, the MERGINGCANDIDATES procedure (see Algorithm 8) is computed, at most, $N-1$ times throughout the entire algorithm, since, starting from the N initial singleton clusters,

the maximum number of new clusters the AHC solution can contain is $N-1$, which occurs when LSS-GCSS algorithm identifies that the dataset is composed of one single hard cluster. The MERGINGCANDIDATES procedure is composed of three different parts.

The update of cluster labels vector $\boldsymbol{\lambda}$ (line 2 in Algorithm 8) involves, regardless of the number of objects belonging to the new cluster, making N comparisons per iteration. Hence, in the worst case ($N-1$ new clusters), its total cost is $O(N^2)$ throughout the entire algorithm.

The determination of the vicinity of every object in the dataset (lines 18–20 in Algorithm 8) entails the assignation of N vicinities (one per object) throughout the entire algorithm. Hence, it involves a N-dependent computation time, regardless of the number of iterations eventually completed.

The rest of Algorithm 8 is dedicated to complete the characterisation of the new cluster ($\mathbf{C}_k$) and entails the computation of a constant (*i.e.* independent from N) number of operations. Considering that there cannot arise more than $N-1$ new clusters, the total cost of this part of MERGINGCANDIDATES procedure is $O(N)$, regardless of the number of iterations eventually completed.

Hence, each one of the, at most, $\frac{N^2-N}{2}$ iterations involves a N-independent computation time, which may vary depending on how the iteration eventually ends (merging of small candidate clusters, merging locally rejected, merging globally rejected, or merging of large candidate clusters). Hence, aside from the calculation, sorting and updating of the proximity values in M_P (lines 3, 12 and 18 in Algorithm 5) and the definition of objects' vicinity (lines 18–20 in Algorithm 8), the total upper bound for the computation time of the LSS-GCSS algorithm once the proximity values in M_P have been calculated and sorted is $\left(\alpha\frac{N^2-N}{2}\right)$-dependent, being α a constant value independent from N.

Therefore, the most restrictive requirement correspond to the sorting of the values in M_P, which lead LSS-GCSS to be an $O(N^2 \log N)$ algorithm in time terms.

4.4 Discussion

While both versions of SL-DID algorithm are based on a single cluster isolation criterion, LSS-GCSS algorithm is derived from the combination of two new cluster merging criteria: LSS can be seen as a result of the improvement of the cluster isolation criterion proposed by Fred and Leitão (2000), whereas GCSS is designed as a complement for LSS in order to better deal with both touching and overlapped clusters. The aim of such foundations is to lead LSS-GCSS both to maintain the upsides of SL-DID (mainly, a behaviour free of tunable parameters, the capability to generate twofold clustering solutions and the ability to handle clusters of all kind) and to be a more flexible algorithm (clusters are never isolated during the agglomeration process), able to

cope with a higher diversity of clustering situations, including those that present touching and overlapped clusters. Furthermore, in spite of involving a higher cost in computation and being a more sophisticated algorithm than SL-DID, the computational requirements of LSS-GCSS keep a quadratic dependency with respect to N both in storage and time terms.

At this stage of the present thesis, the real benefits, if any, of using LSS-GCSS algorithm instead of other both partitional and hierarchical clustering algorithms need to be properly measured and determined. The next chapter in the present thesis is therefore focused on evaluating the performance results of LSS-GCSS algorithm on different datasets of varied nature, defining its limitations and determining in what measure it involves an improvement with respect to both SL-DID and other clustering algorithms.

Chapter 5

Experimental performance of LSS-GCSS algorithm

The main contribution of the present thesis has been presented in the previous chapter. Derived from on the combination of LSS and GCSS cluster merging criteria, the LSS-GCSS algorithm aims to overcome the limitations of the previous approaches to parameter-free AHC clustering and to be able to successfully deal with a variety of clustering situations as wide and diverse as possible: randomly generated data, real-word clustering scenarios, clusters of different shapes and distributions, datasets with different density regions, touching and overlapped clusters, balanced and unbalanced clusters, single- and multiple-cluster datasets, low- and high-dimensional data, data of different origin and nature (*e.g.* text, speech, image), etc. Therefore, the contributions of the present chapter derive from an exhaustive evaluation of the performance of LSS-GCSS in the face of a great diversity of clustering scenarios, so that both upsides and limitations that characterise the behaviour of the LSS-GCSS algorithm are defined. Additionally, this evaluation includes a comparative study among a variety of clustering algorithms with the aim of determining to what extent LSS-GCSS outperforms the capabilities of other clustering methods.

Thus, this fifth chapter is structured as follows. The general setup of the experiments is described in section 5.1. Next, sections 5.2 and 5.3 include all the experiments relative to the study of the performance of LSS-GCSS algorithm in the face of both synthetic and real data, respectively. A comparison of performance among a diversity of clustering algorithms is provided in section 5.4. And finally, conclusions and considerations about the obtained experimental results are detailed in section 5.5, which includes a discussion directly referred to both the first three research questions and the first research hypothesis posed in section 1.3.

5.1 Experimental setup

After the design and implementation of the LSS-GCSS algorithm (see Chapter 4), the next contribution of the present thesis consists of a detailed study of its performance, which is evaluated in a wide variety of clustering scenarios. Mainly, this study is carried out in order to achieve the following two goals:

1. To gain an awareness of both profits and limitations of LSS-GCSS algorithm, which may be useful in order to decide, depending on what prior knowledge about the clustering scenario is available, how to make use of LSS-GCSS in practice.
2. To compare LSS-GCSS with other clustering algorithms in terms of performance, which may lead to relevant information in order to decide, given a specific clustering problem, what clustering algorithm best fits the characteristics of the scenario.

Hence, with the aim of accomplishing such goals and obtaining both reliable and profitable conclusions, a complete set of experiments is defined and arranged throughout the following three sections, each comprising a diversity of datasets and clustering situations:

- **Synthetic datasets** (see section 5.2 for further details). In the same way that other related works in the literature (Fred and Leitão, 2003; Aidos and Fred, 2011a), a wide set of different synthetic datasets –artificially created, not belonging to real clustering scenarios– is created in order to have control over some specific parameters (*e.g.* number of clusters, data dimensionality, closeness/overlapping between clusters, distribution of objects within clusters, balancing between clusters, shape of clusters, etc.). This strategy allows to evaluate how the performance of LSS-GCSS varies according to determinate characteristics of the data.
- **Real datasets** (see section 5.3 for further details). As a complement to the experiments performed on synthetic datasets, LSS-GCSS algorithm is also tested on a set of well-known real datasets, which come from real-world scenarios and include data from different origin and nature. These benchmark datasets are widely used in the literature in order both to evaluate how clustering algorithms behave when dealing with real clustering scenarios and to compare their performance under equal and known conditions.
- **Comparative study** (see section 5.4 for further details). A set of both synthetic and real datasets is selected from the two previous sections in order to conduct a comparison between LSS-GCSS and other clustering algorithms (both partitional and hierarchical, both parameter-dependent and parameter-free) in terms of performance.

It is worth noting that the ground truth solution is known in advance for all the datasets tested in this chapter, which includes both their optimal number of clusters (K_{opt}) and their optimal cluster

labels. Since all the cluster analyses in the present chapter are performed in a blind manner (*i.e.* without considering any prior knowledge about the structure of data), this information is only used for evaluation purposes, in order to determine the validity of the obtained clustering results (see section 2.4).

5.2 Synthetic datasets

The main goal of the present section is to study in what measure the performance of the LSS-GCSS algorithm depends on some specific characteristics of the data. To that effect, a great diversity of synthetic datasets is created and arranged throughout the following clustering scenarios:

- **Single-cluster datasets** (see section 5.2.1 for further details). Does LSS-GCSS tend to identify clusters where there are none? How does LSS-GCSS behave when the dataset comprises a single cluster only?

- **Touching and overlapped clusters** (see section 5.2.2 for further details). Does LSS-GCSS have the ability to identify both touching and overlapped clusters? Which degree of closeness/overlapping between two different clusters is LSS-GCSS able to admit?

- **Unbalanced clusters** (see section 5.2.3 for further details). Does LSS-GCSS have the ability to identify unbalanced clusters? Which degree of unbalancing between two different clusters is LSS-GCSS able to admit?

- **Multiple-cluster datasets** (see section 5.2.4 for further details). Does LSS-GCSS have difficulties in dealing with a high amount of clusters?

- **Concentric clusters** (see section 5.2.5 for further details). Does LSS-GCSS have the ability to identify concentric clusters (typically troublesome clusters for centre-based clustering algorithms)?

- **Arbitrary-shaped clusters** (see section 5.2.6 for further details). Does LSS-GCSS have the ability to identify clusters regardless of their shape?

All the datasets defined in this section present real numerical features and the proximity between objects is measured by means of the Euclidean distance (see equation 2.8) in all the clustering scenarios. Furthermore, aside from comparing the number of identified clusters (K) with the real number of clusters in the ground truth solution (K_{opt}), every clustering solution obtained in the present section is externally validated by means of the Consistency Index (CI) (see section 2.4.1 for further details).

5.2.1 Single-cluster datasets

Input data

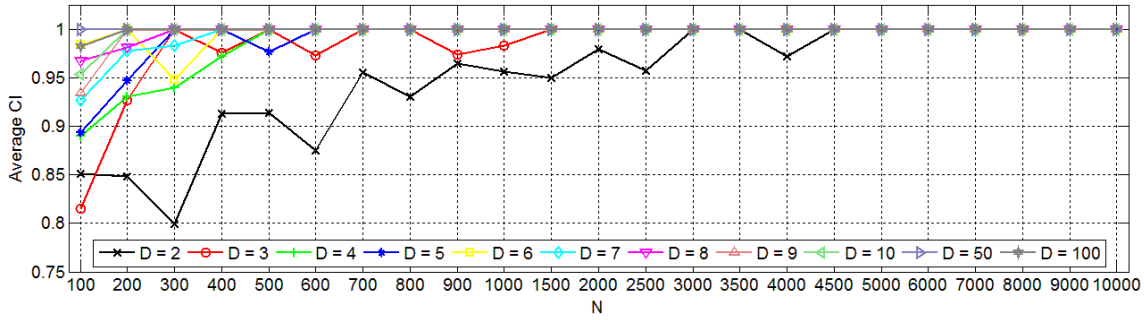
It is typical for clustering algorithms to tend to find clusters regardless of whether they actually exist or not, eventually being able to impose inappropriate clustering structures on unstructured or random data (Fred and Leitão, 2003). Hence, a test for absence of cluster structure prior to the cluster analysis may be required (Dubes and Jain, 1979; Everitt et al., 2011). The behaviour of LSS-GCSS when handling single-cluster datasets is studied in the present section. To that end, *1Gauss* and *1unif* datasets are defined. They both comprise one single D-dimensional cluster ($K_{opt} = 1$) of N objects randomly generated from Gaussian and uniform distributions, respectively.

Characterisation

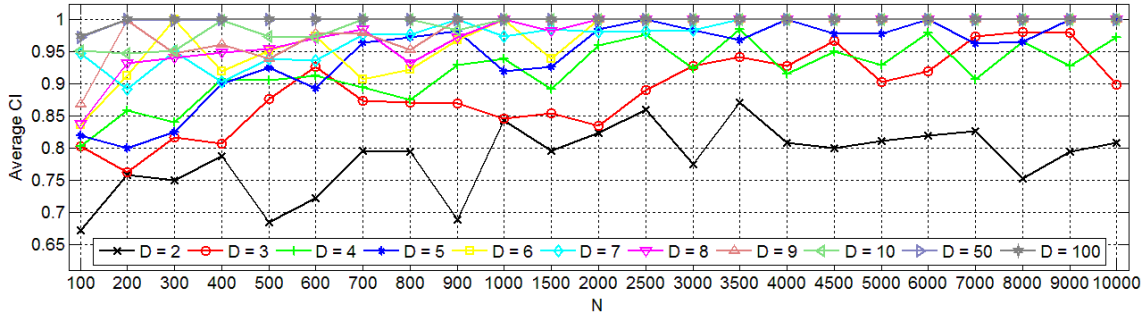
- The study covers multiple values of both the number of objects ($N \in \{100, 10000\}$) and the data dimensionality ($D \in \{2, 100\}$) per every dataset.
- 100 instances of both datasets are tested per each combination of parameters N and D.

Cluster analysis

Figures 5.1, 5.2 and 5.3 show the performance of LSS-GCSS in terms of both CI and K.

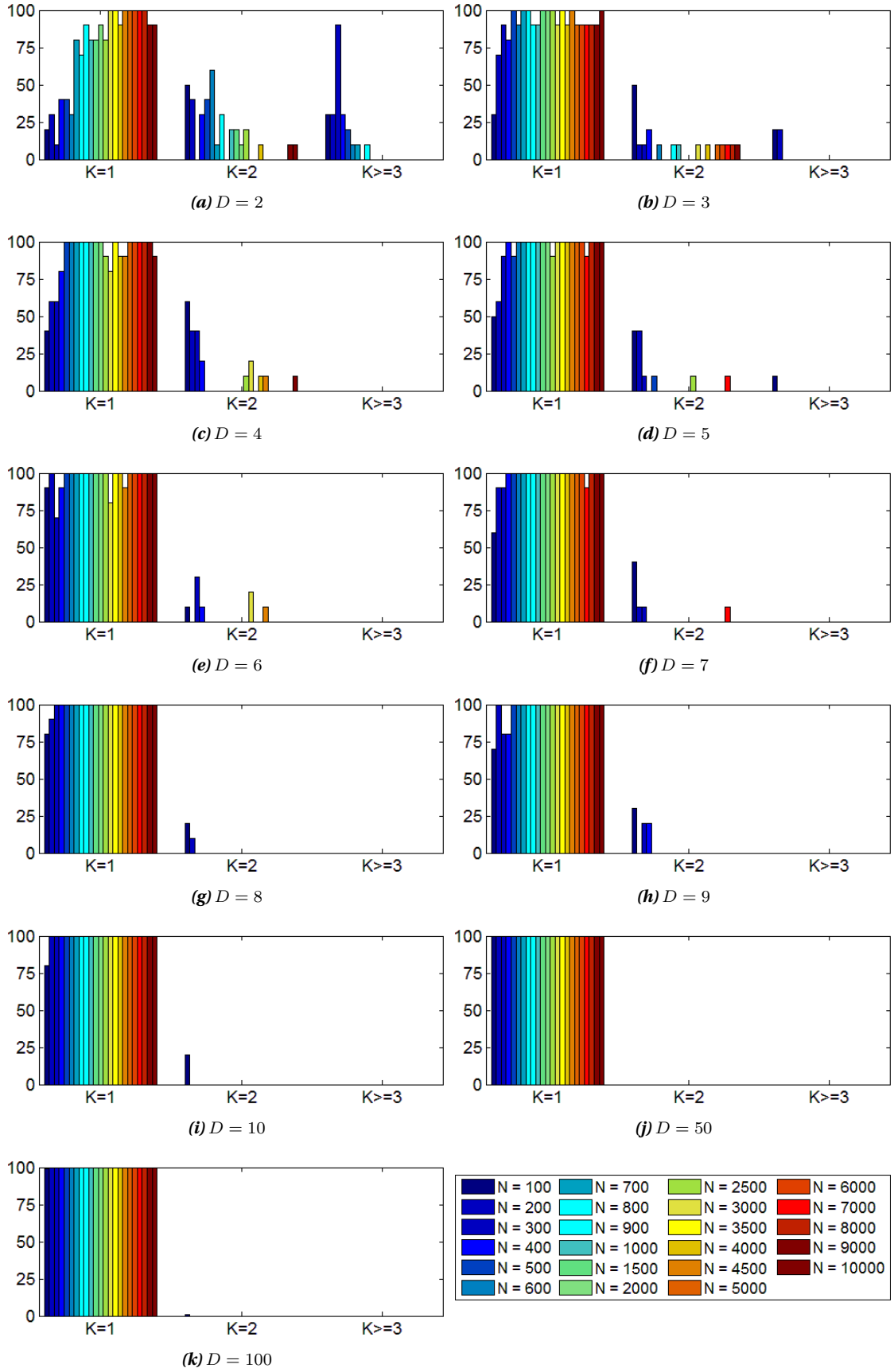


(a) *1Gauss* dataset: CI averaged-values obtained by LSS-GCSS.



(b) *1unif* dataset: CI averaged-values obtained by LSS-GCSS.

Figure 5.1: Performance of LSS-GCSS on single-cluster datasets. Results are averaged along 100 instances.



(a) $D = 2$ ***(b)*** $D = 3$

(c) $D = 4$ ***(d)*** $D = 5$

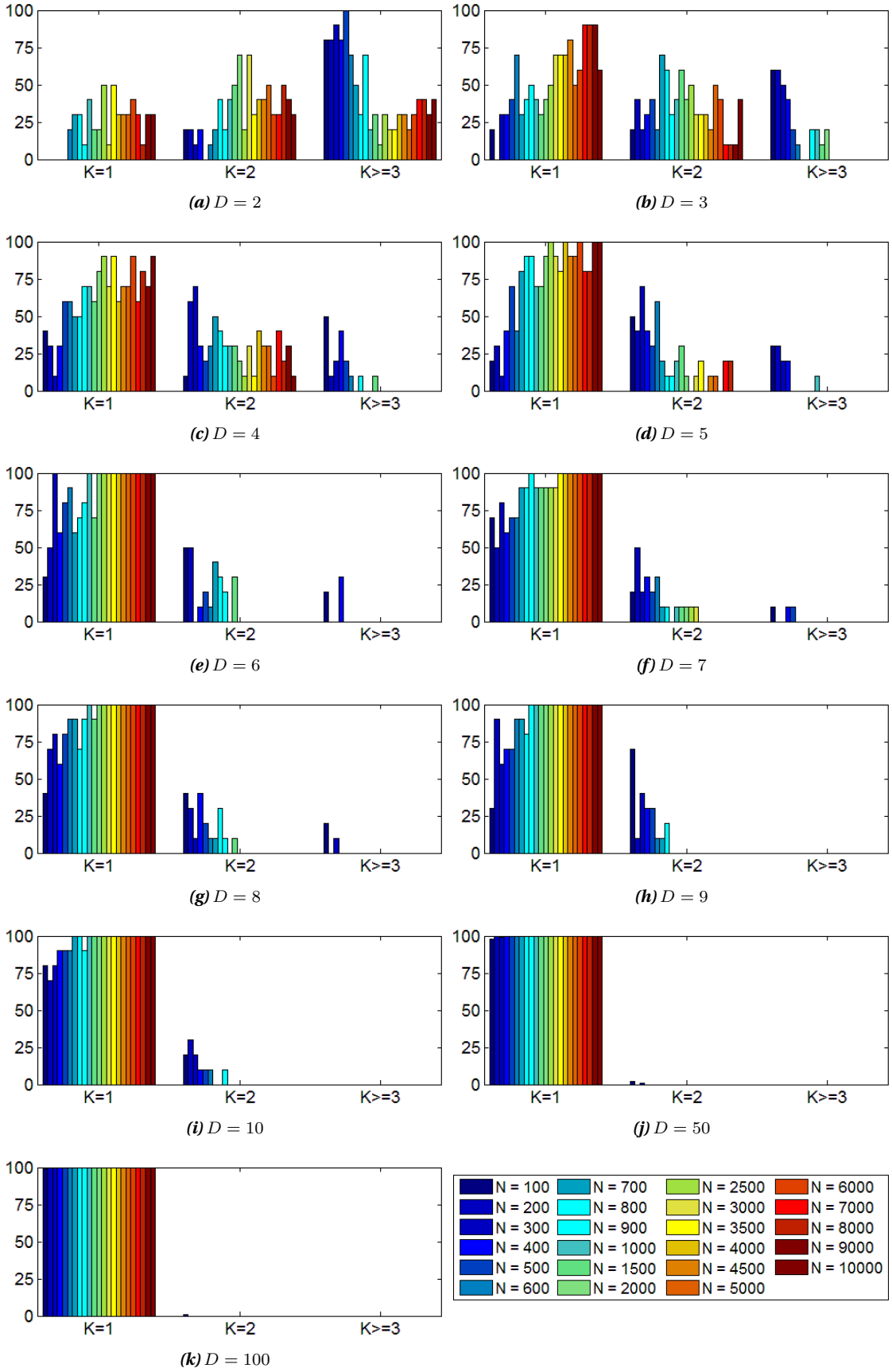
(e) $D = 6$ ***(f)*** $D = 7$

(g) $D = 8$ ***(h)*** $D = 9$

(i) $D = 10$ ***(j)*** $D = 50$

(k) $D = 100$

Figure 5.2: Performance of LSS-GCSS on *1Gauss* dataset: histograms of K. Every histogram corresponds to a specific value of D and includes the 100 instances corresponding to each value of N.



(a) $D = 2$ ***(b)*** $D = 3$

(c) $D = 4$ ***(d)*** $D = 5$

(e) $D = 6$ ***(f)*** $D = 7$

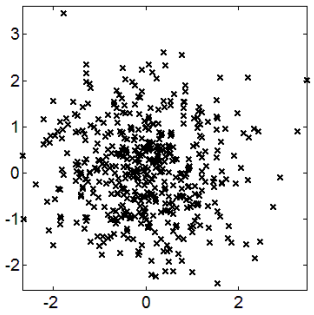
(g) $D = 8$ ***(h)*** $D = 9$

(i) $D = 10$ ***(j)*** $D = 50$

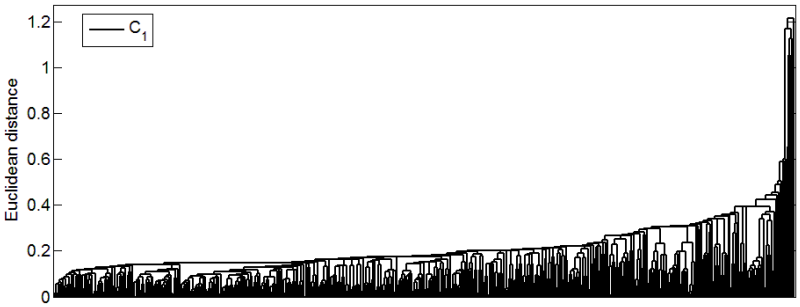
(k) $D = 100$

Figure 5.3: Performance of LSS-GCSS on *1unif* dataset: histograms of K. Every histogram corresponds to a specific value of D and includes the 100 instances corresponding to each value of N.

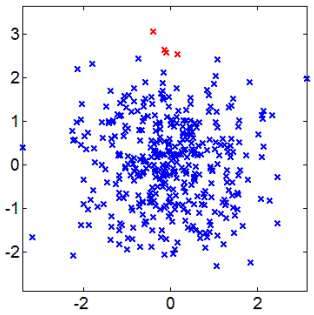
Moreover, examples of clustering solutions obtained by LSS-GCSS on two-dimensional ($D = 2$) instances of both *1Gauss* and *1unif* datasets are shown in Figure 5.4:



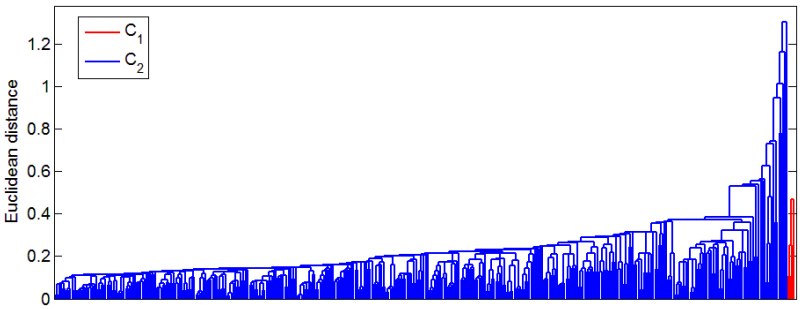
(a) *1Gauss* dataset: scatterplot.



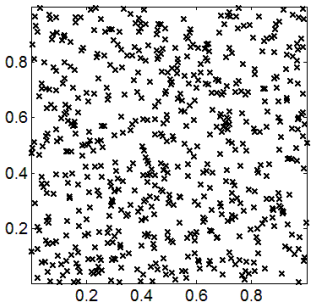
(b) *1Gauss* dataset: LSS-GCSS dendrogram ($N = 500,\ D = 2,\ K = 1,\ CI = 1$).



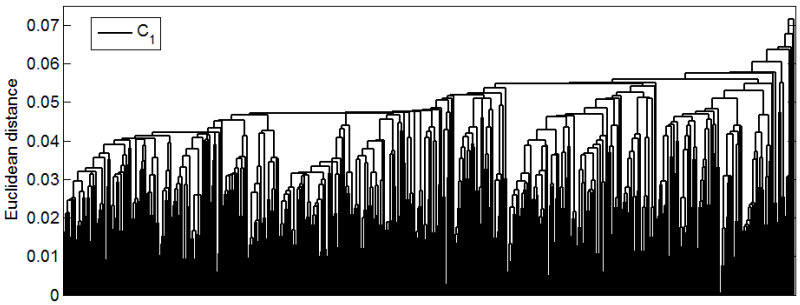
(c) *1Gauss* dataset: scatterplot.



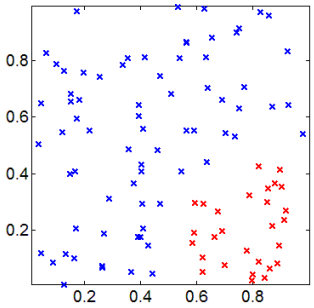
(d) *1Gauss* dataset: LSS-GCSS dendrograms ($N = 400,\ D = 2,\ K = 2,\ CI = 0.988$).



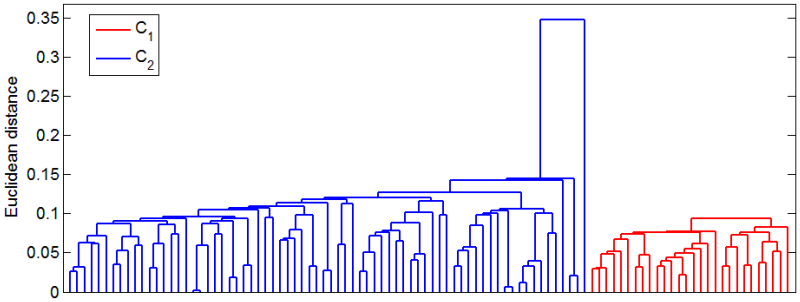
(e) *1unif* dataset: scatterplot.



(f) *1unif* dataset: LSS-GCSS dendrogram ($N = 600,\ D = 2,\ K = 1,\ CI = 1$).



(g) *1unif* dataset: scatterplot.



(h) *1unif* dataset: LSS-GCSS dendrograms ($N = 400,\ D = 2,\ K = 2,\ CI = 0.72$).

Figure 5.4: Single-cluster datasets: clustering solutions by LSS-GCSS.

Results

In general terms, the evaluation of the clustering results shown in Figures 5.1, 5.2 and 5.3 indicates that LSS-GCSS tends to successfully identify one single cluster to the extent that both the num-

ber of objects (N) and the dimensionality (D) grow. Furthermore, while this behaviour stands regardless of the cluster distribution, it is also noticeable that, regarding single-cluster datasets, LSS-GCSS tends to behave more accurately when dealing with Gaussian distributed clusters than with uniformly distributed ones.

In addition, more specific conclusions can be drawn:

- LSS-GCSS tends to always identify one single Gaussian cluster $(K = 1)$ when the number of objects is large enough $(N \geq 3000)$, regardless of the data dimensionality (see Figures 5.1a and 5.2).
- To reach stable successful results $(K = 1)$ when handling one single uniformly-distributed cluster, LSS-GCSS requires, in addition, a data dimensionality large enough $(D \geq 6)$ (see Figures 5.1b and 5.3).
- Due to the nature of both distributions (unlike the uniform distribution, the Gaussian distribution presents outlier objects), clustering errors in *1Gauss* dataset tend to affect smaller amounts of objects than in *1unif* dataset, therefore leading to less-penalised CI values (see Figures 5.4c and 5.4g).
- Figure 5.4f clearly shows how the LSS-GCSS dendrograms of uniform clusters are formed: objects join together first in small subclusters, which in their turn progressively merge into a larger cluster. This agglomeration tendency justifies why the GCSS criterion requires a balance factor (see equation 4.14) in order to be able to successfully handle uniformly distributed clusters: a small group of objects in comparison with the rest of the cluster, but not with the other small subclusters already merged, can break the agglomeration tendency in the last stages of the agglomeration process and be misidentified as an actual different cluster.

  Moreover, this undesired behaviour justifies why LSS-GCSS deals better with a single Gaussian cluster than with a uniformly distributed one. It also explains why low-populated two-dimensional instances of the *1unif* dataset present the poorest clustering results, since singular groups of objects are easier to arise in such an scenario (see Figure 5.4h).
- It is also worth noting how LSS-GCSS dendrograms may be helpful to visually identify how objects are distributed within their clusters: the progressive increment of the proximities in the dendrogram shown in Figure 5.4b indicates the presence of a variable density of objects within the cluster (which matches with the nature of the Gaussian distribution) and contrasts with the more homogeneous proximity values present in the dendrogram of Figure 5.4f, which suggest a more constant density of objects (typical of a uniform distribution).

Finally, the tendency of LSS-GCSS to improve its performance as the data dimensionality grows has an explanation in the "curse of dimensionality" (Bellman, 1961): the volume of the space increases fast with the dimensionality and, therefore, data becomes sparse, which is problematic when

statistical significance is required. Among other effects, this phenomenon causes that the proximity between any two objects in a high-dimensional space becomes practically the same (Beyer et al., 1999). This leads to an homogenisation of the values in the proximity matrix (see equation 2.12), which brings about an absence of significantly high gaps in the resulting dendrogram. In such conditions, the higher the data dimensionality, the easier for LSS-GCSS (as well as for both any other clustering algorithm and any test for absence of cluster structure) to conclude that all objects in the dataset are grouped into one single cluster.

5.2.2 Touching and overlapped clusters

The main drawback of SL-DID algorithm (the clustering algorithm LSS-GCSS arises from) is its lack of performance in the face of both touching and overlapped clusters (see section 3.3). Thus, the performance of LSS-GCSS algorithm when dealing with datasets that present touching and overlapped clusters is studied in the present section, which is structured as follows:

- The study on how LSS-GCSS handles touching clusters is carried out in section 5.2.2.1 by means of the *2unif* dataset.
- The performance of LSS-GCSS in the face of the *2Gauss* dataset (probably, the most typical case of overlapped clusters present in the literature) is tackled in section 5.2.2.2.
- A particular case of touching/overlapped clusters (the *2bars* dataset) is studied in section 5.2.2.3.

5.2.2.1 The *2unif* dataset

Input data

The *2unif* dataset comprises one thousand objects ($N = 1000$) structured into two uniformly distributed D-dimensional clusters ($K_{opt} = 2$). Both clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) are perfectly balanced ($N_1 = N_2 = 500$) and separated a distance d along their frontier region: whereas objects belonging to $\mathbf{C}_1$ are located between 0 and 1 along all dimensions $\left(x_{i_j} \in [0,1]\,, \forall\, \mathbf{x}_i \in \mathbf{C}_1, \forall\, j \in \{1, D\}\right)$, objects belonging to $\mathbf{C}_2$ are moved along the first dimension ($x_{i_1} \in [1+d,\, 2+d]\,, \forall\, \mathbf{x}_i \in \mathbf{C}_2$) and located between 0 and 1 along the rest of dimensions $\left(x_{i_j} \in [0,1]\,, \forall\, \mathbf{x}_i \in \mathbf{C}_2, \forall\, j \in \{2, D\}\right)$.

Characterisation

- The study covers multiple values of both the distance between clusters ($d \in [0,2]$) and the data dimensionality ($D \in \{2, 100\}$).
- 100 instances of the *2unif* dataset are tested per each combination of parameters d and D.

Cluster analysis

Firstly, the performance of LSS-GCSS in terms of both CI and K is shown in Figures 5.5 and 5.7. Moreover, examples of clustering solutions obtained by LSS-GCSS on two-dimensional $(D = 2)$ instances of the *2unif* dataset are shown in Figure 5.6.

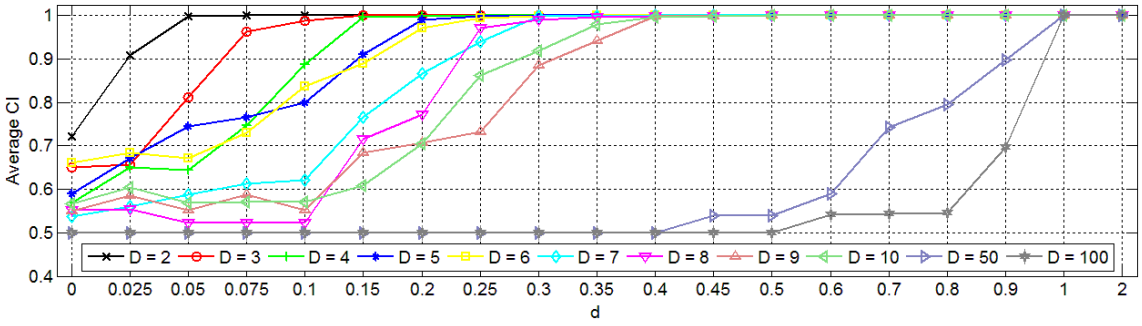
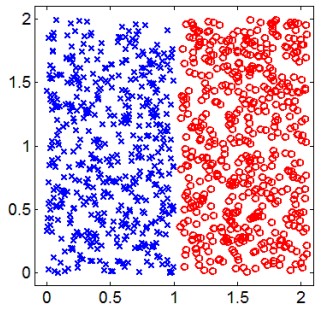
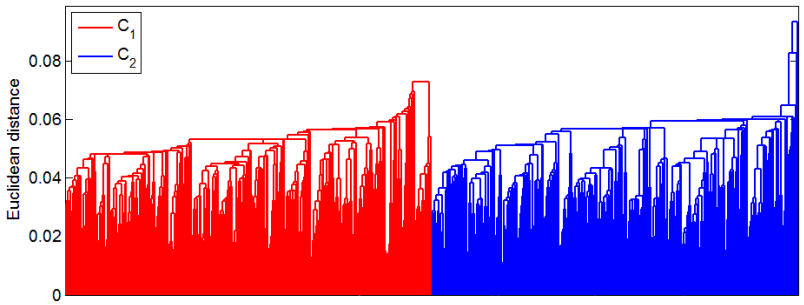


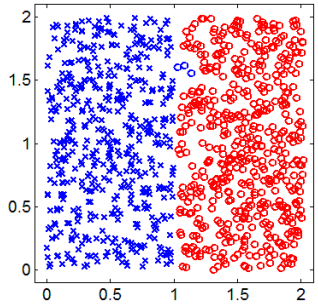
Figure 5.5: Performance of LSS-GCSS on *2unif* dataset. CI values averaged along 100 instances.



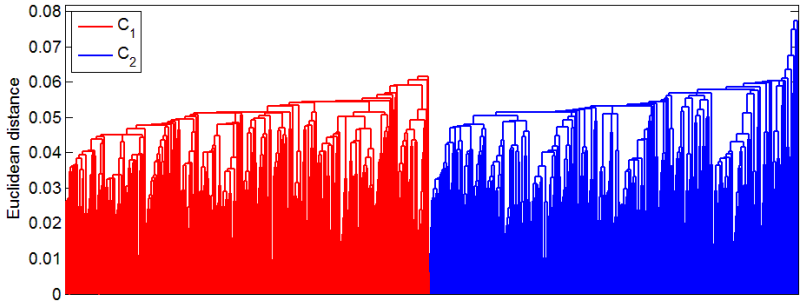
(a) *2unif* dataset: scatterplot.



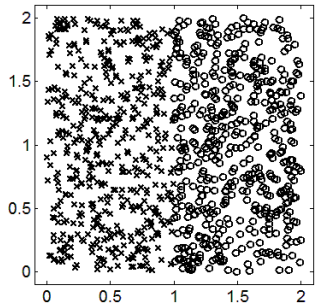
(b) *2unif* dataset: LSS-GCSS dendrograms $(d = 0.05,\ D = 2,\ K = 2,\ CI = 1)$.



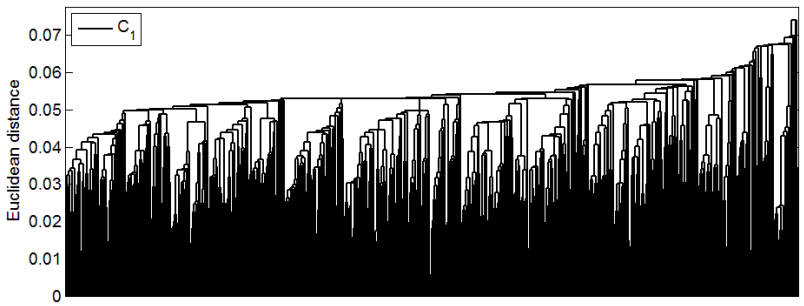
(c) *2unif* dataset: scatterplot.



(d) *2unif* dataset: LSS-GCSS dendrograms $(d = 0.025,\ D = 2,\ K = 2,\ CI = 0.997)$.

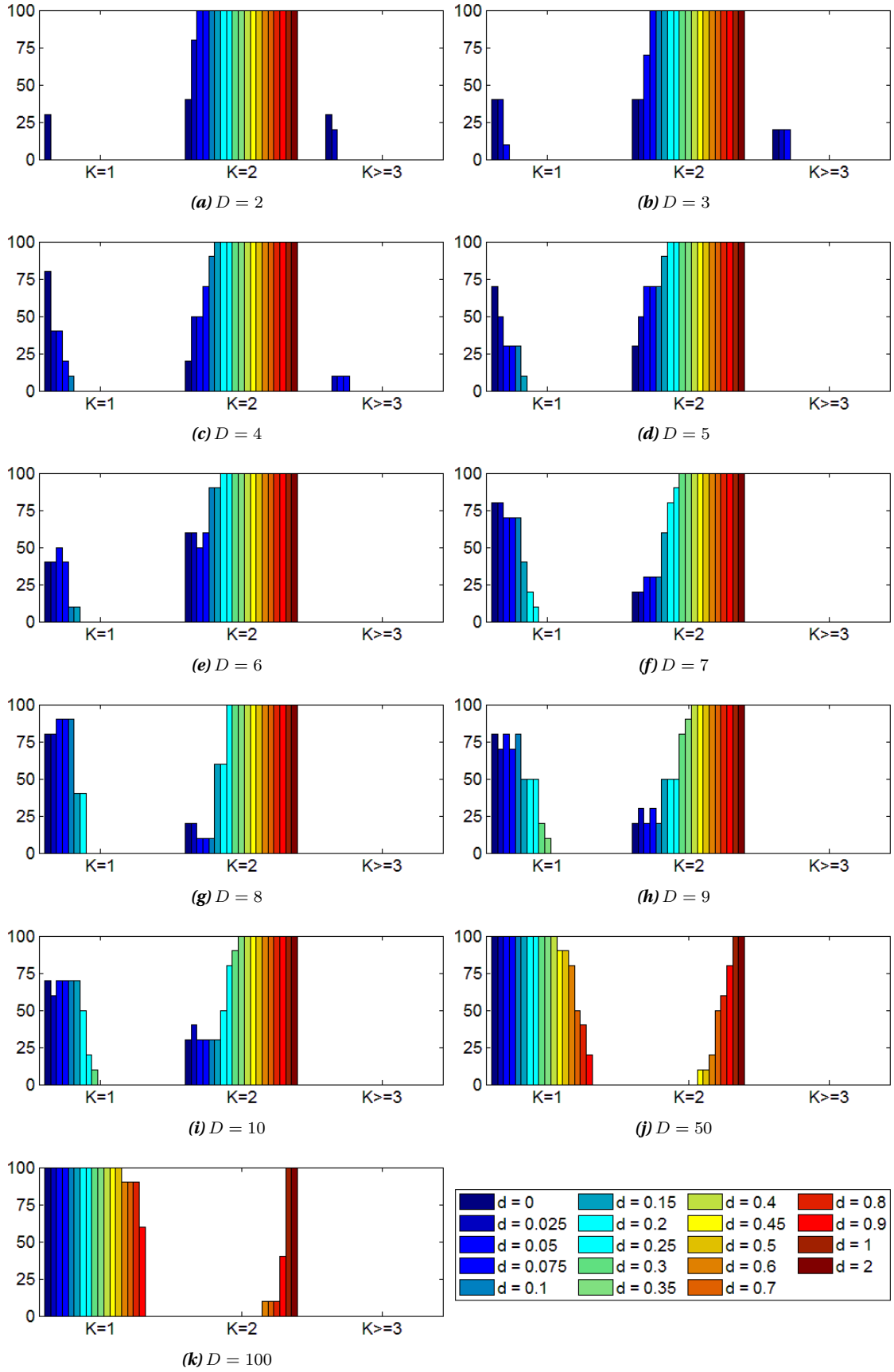


(e) *2unif* dataset: scatterplot.



(f) *2unif* dataset: LSS-GCSS dendrogram $(d = 0,\ D = 2,\ K = 1,\ CI = 0.5)$.

Figure 5.6: *2unif* dataset: clustering solutions by LSS-GCSS.



(a) $D = 2$ ***(b)*** $D = 3$

(c) $D = 4$ ***(d)*** $D = 5$

(e) $D = 6$ ***(f)*** $D = 7$

(g) $D = 8$ ***(h)*** $D = 9$

(i) $D = 10$ ***(j)*** $D = 50$

(k) $D = 100$

Figure 5.7: Performance of LSS-GCSS on *2unif* dataset: histograms of K. Every histogram corresponds to a specific value of D and includes the 100 instances corresponding to each value of d.

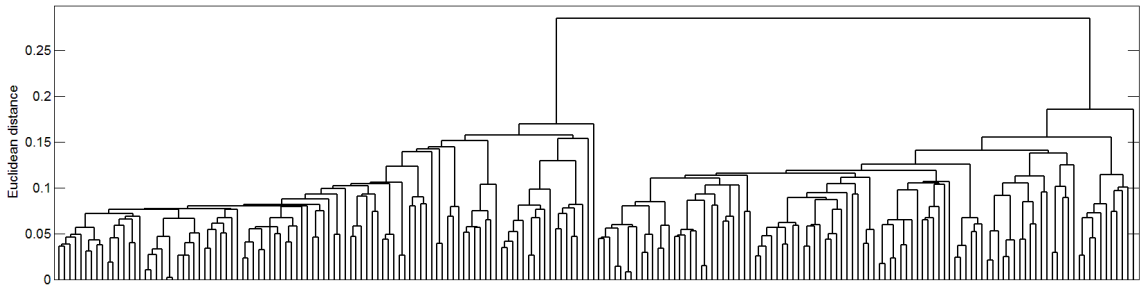
Results

The clustering results illustrated in Figures 5.5 and 5.7 indicate that LSS-GCSS successfully deals with touching clusters, even for remarkably short separation distances. Two-dimensional clustering results $(D = 2)$ show that LSS-GCSS begins to not distinguish between both clusters only when the frontier region becomes imperceptible $(d < 0.05)$ (see Figure 5.6). Furthermore, the effects of the dimensionality can be also appreciated: wider frontier regions are required in order to identify two clusters as the dimensionality grows.

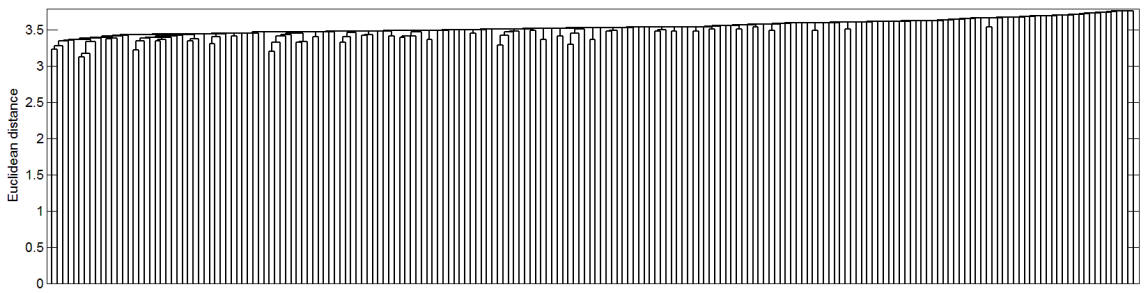
In addition, more specific conclusions can be drawn:

- LSS-GCSS identifies more than two clusters $(K \geq 3)$ only in a few low-dimensional instances $(D \leq 4)$ of the *2unif* dataset (see Figures 5.7a, 5.7b and 5.7c). Despite of being formed by uniformly distributed objects (which, as shown in section 5.2.1, are more likely to be grouped into more than one cluster by LSS-GCSS), the size of the clusters in the *2unif* dataset is always lower in relative terms (with respect to the total size of the dataset) than in any single-cluster dataset. This fact increases the effect of the size factors of both LSS and GCSS criteria, which, in addition with the effect of the balance factor of the GCSS criterion, cause this desirable result.

- It is worth noting why the clustering results tends now to worsen due to the effect of dimensionality. Assuming that the variance $\left(x_{i_j} \in [0,1]\right)$ and the size $(N_1$ and $N_2)$ of both clusters remain constant, objects become more sparse within their own clusters as the dimensionality (D) grows, so that the proximities between the objects belonging to the same cluster go



(a) *2unif* dataset: SL-dendrogram $(K_{opt} = 2,\ N_1 = N_2 = 100,\ d = 0.25,\ D = 2)$.



(b) *2unif* dataset: SL-dendrogram $(K_{opt} = 2,\ N_1 = N_2 = 100,\ d = 0.25,\ D = 100)$.

Figure 5.8: *2unif* dataset: SL-dendrograms.

up. Since the separation between both clusters (d) does not depend on the data dimensionality, the differences between intra-cluster proximities and the distance between both clusters diminishes as dimensionality grows. Hence, both clusters eventually become indistinguishable when D is high enough, unless the value of d is also increased (see Figure 5.5).

This effect is clearly illustrated in Figure 5.8, where the SL-dendrograms of two instances of the *2unif* dataset are shown. Despite the fact that all the parameters of the dataset remain constant except for the data dimensionality, the differences between both dendrograms are noticeable: the significantly high final proximity ($d \approx 0.25$) in the dendrogram of the 2-dimensional instance clearly suggests the presence of two well-separated clusters (see Figure 5.8a), whereas the dendrogram of the 100-dimensional instance indicates that all objects are contained in one single cluster (see Figure 5.8b). The effect of dimensionality is clearly illustrated by the range of proximities covered by the second dendrogram ($d \in [3.13, 3.766]$), which is much higher than the separation distance between both distributions ($d = 0.25$). A higher distance between clusters ($d \geq 1$) is required under such conditions in order to obtain truly separated clusters (see $D = 50$ and $D = 100$ plots in Figure 5.5).

- The performance of LSS-GCSS has been also tested in the face of differently sized instances of the *2unif* dataset ($N \in \{200, 10000\}$) and the obtained clustering results have no suffered from any significant variation. Hence, the present conclusions and considerations can be generalised for the *2unif* dataset, regardless of the size of its clusters.

5.2.2.2 The *2Gauss* dataset

Input data

The *2Gauss* dataset comprises one thousand objects ($N = 1000$) structured into two Gaussian distributed D-dimensional clusters ($K_{opt} = 2$). Both clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) are perfectly balanced ($N_1 = N_2 = 500$). Like any Gaussian distribution, both clusters are completely characterised by their respective means (μ_1 and μ_2) and standard deviations (σ_1 and σ_2). They both have unitary variance ($\sigma_1 = \sigma_2 = \sigma = 1$) and their degree of overlapping is ruled over by the distance between their means (d), which is measured in units of σ ($d = n\,\sigma$, $\forall\, n \in \mathbb{R}^+$): while $n = 2$ ($d = 2\,\sigma$) causes a high overlapping between $\mathbf{C}_1$ and $\mathbf{C}_2$, $n = 6$ ($d = 6\,\sigma$) leads to faintly overlapped clusters.

Characterisation

- The study covers multiple values of both the overlapping degree between clusters ($n \in [2, 10]$) and the data dimensionality ($D \in \{2, 100\}$).
- 100 instances of the *2Gauss* dataset are tested per each combination of parameters n and D.

Cluster analysis

Firstly, the performance of LSS-GCSS in terms of both CI and K is shown in Figures 5.9 and 5.11. Moreover, examples of clustering solutions obtained by LSS-GCSS on two-dimensional $(D = 2)$ instances of the *2Gauss* dataset are shown in Figure 5.10.

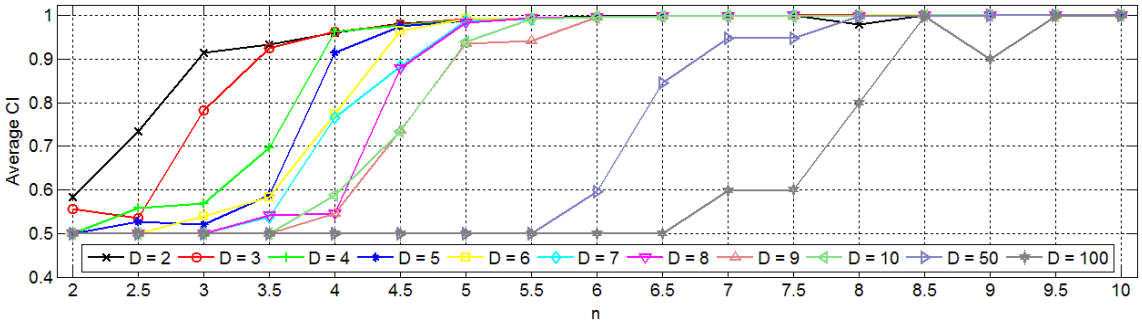
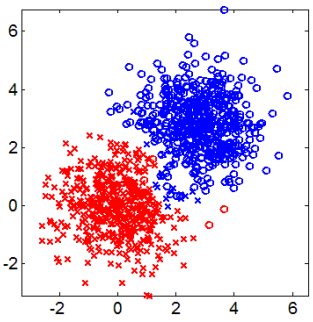
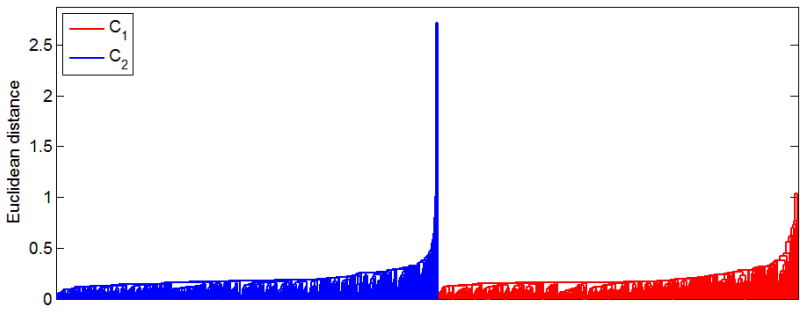


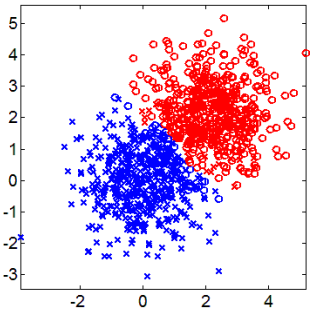
Figure 5.9: Performance of LSS-GCSS on *2Gauss* dataset. CI values averaged along 100 instances.



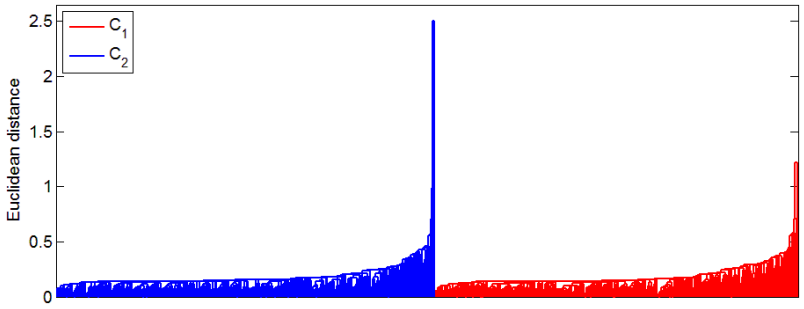
(a) *2Gauss* dataset: scatterplot.



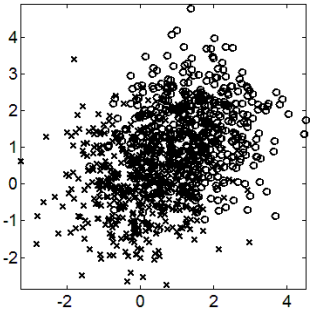
(b) *2Gauss* dataset: LSS-GCSS dendrograms ($n = 4,\ D = 2,\ K = 2,\ CI = 0.975$).



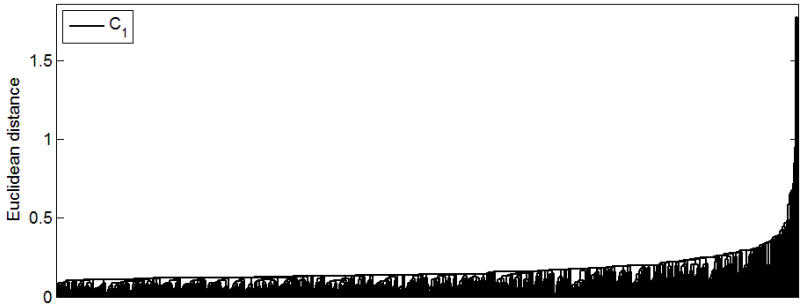
(c) *2Gauss* dataset: scatterplot.



(d) *2Gauss* dataset: LSS-GCSS dendrograms ($n = 3,\ D = 2,\ K = 2,\ CI = 0.932$).

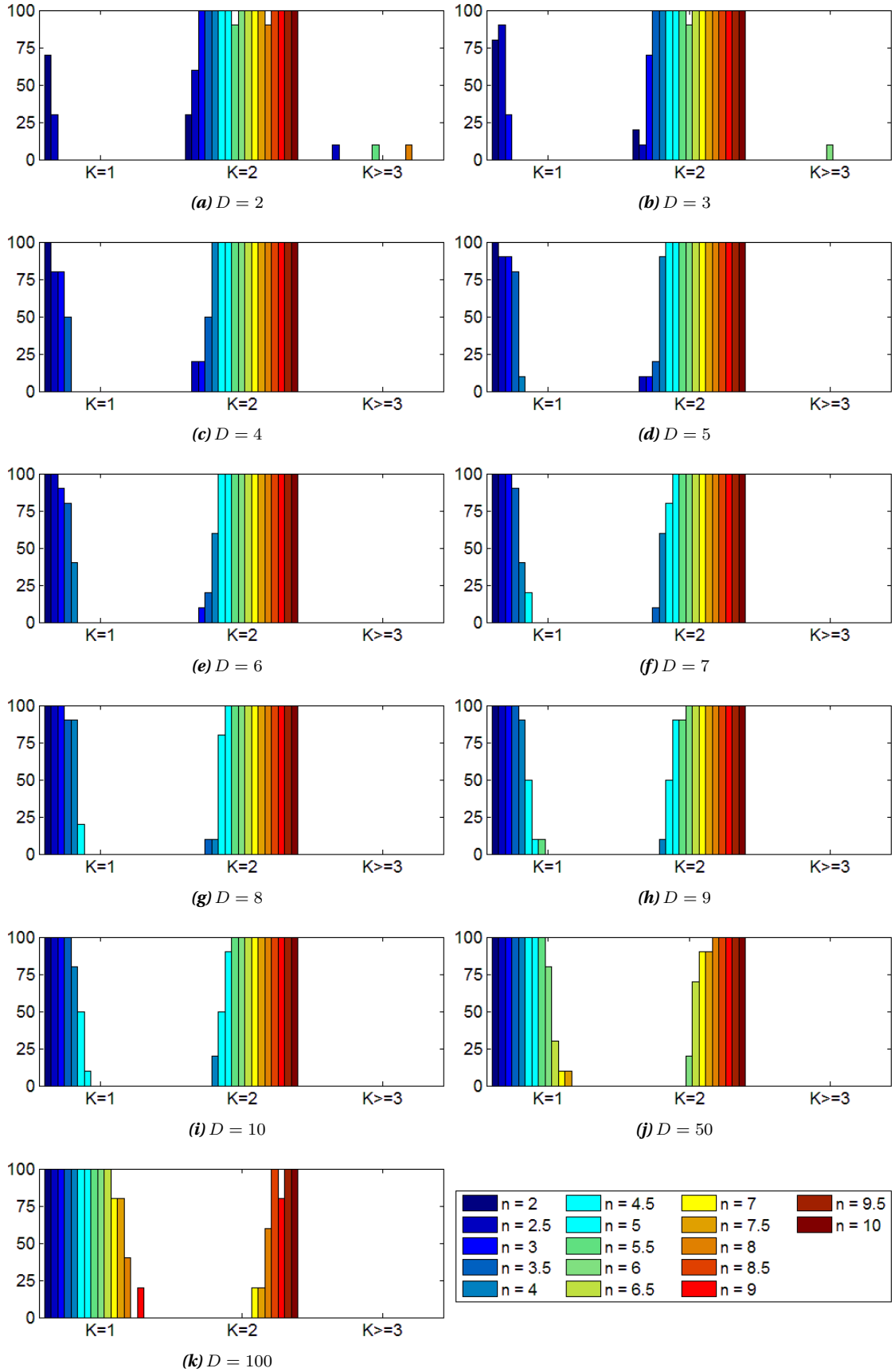


(e) *2Gauss* dataset: scatterplot.



(f) *2Gauss* dataset: LSS-GCSS dendrogram ($n = 2,\ D = 2,\ K = 1,\ CI = 0.5$).

Figure 5.10: *2Gauss* dataset: clustering solutions by LSS-GCSS.



(a) $D = 2$ ***(b)*** $D = 3$

(c) $D = 4$ ***(d)*** $D = 5$

(e) $D = 6$ ***(f)*** $D = 7$

(g) $D = 8$ ***(h)*** $D = 9$

(i) $D = 10$ ***(j)*** $D = 50$

(k) $D = 100$

Figure 5.11: Performance of LSS-GCSS on *2Gauss* dataset: histograms of K. Every histogram corresponds to a specific value of D and includes the 100 instances corresponding to each value of n.

Results

The clustering results illustrated in Figures 5.9 and 5.11 indicate that LSS-GCSS successfully deals with overlapped clusters, even for strong degrees of overlapping. Two-dimensional clustering results ($D = 2$) show that LSS-GCSS begins to not distinguish between both clusters only when the degrees of overlapping becomes remarkably intense ($n < 3$) (see Figure 5.11a). Again, the effects of the dimensionality affect the clustering results: distance between means (d) has to increase in order for LSS-GCSS to successfully distinguish between both clusters as the dimensionality grows.

It is worth noting that only when both clusters are completely non-overlapped, perfect clustering results ($CI = 1$) can be reached. Nonetheless, considering the two-dimensional instances by way of example, LSS-GCSS obtains certainly accurate results ($K = 2$) when clusters are clearly overlapped ($d \leq 4$) without incurring into a perfect grouping of all objects in the dataset ($CI < 1$) (see Figure 5.10).

Finally, considerations drawn in the case of the *2unif* dataset (*i.e.* the few low-dimensional instances where LSS-GCSS identifies more than two clusters, the reasons according to which the effect of dimensionality worsens the clustering results and the generalisation of the results regardless of the size of the clusters) apply equally to the *2Gauss* dataset (see section 5.2.2.1).

5.2.2.3 The *2bars* dataset

Input data

The *2bars* dataset comprises a slightly random amount ($N \in \{1314, 1373\}$) of 2-dimensional objects ($D = 2$) grouped into two clusters ($K_{opt} = 2$), whose density of objects progressively diminishes along the first dimension of the dataset. Both clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) are faintly unbalanced ($N_1 \in \{642, 706\}$, $N_2 \in \{651, 705\}$) and they both can be either touching or overlapped clusters. Whereas the location of $\mathbf{C}_1$ is static ($x_{i_1} \in [0, 1]$, $x_{i_2} \in [0, 2]$, $\forall\, \mathbf{x}_i \in \mathbf{C}_1$), the location of $\mathbf{C}_2$ varies along the first dimension of the dataset ($x_{i_1} \in [1 + \Delta_x, 2 + \Delta_x]$, $x_{i_2} \in [0, 2]$, $\forall\, \mathbf{x}_i \in \mathbf{C}_2$).

Thus, short shiftings ($\Delta_x \geq -0.15$) lead to touching clusters, while long shiftings ($\Delta_x < -0.15$) cause clusters to be overlapped. It is also noticeable that the frontier between clusters is delimited by a high-density region and a low-density region, each one belonging to a different cluster.

Characterisation

- The study covers multiple values of the distance between clusters, which is adjusted by the shifting of $\mathbf{C}_2$ along the first dimension of the dataset ($\Delta_x \in [-1, 0]$).
- 100 instances of the *2bars* dataset are tested per each value of parameter Δ_x.

Cluster analysis

Figures 5.12 and 5.13 show the performance of LSS-GCSS in terms of both CI and K.

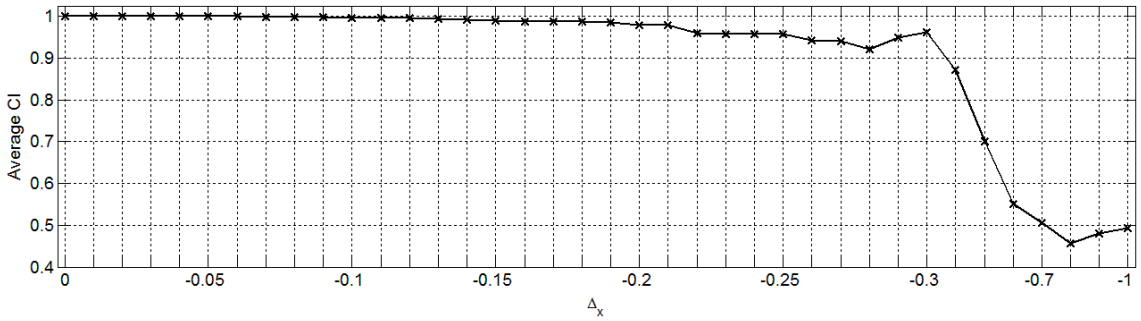


Figure 5.12: Performance of LSS-GCSS on *2bars* dataset. CI values averaged along 100 instances.

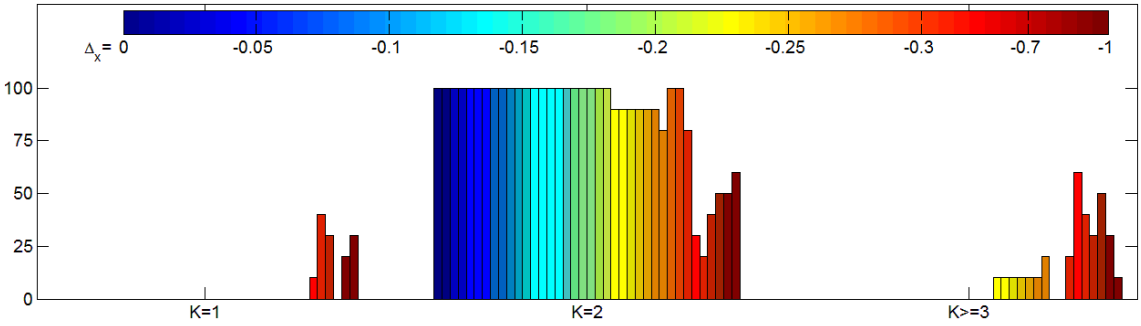


Figure 5.13: Performance of LSS-GCSS on *2bars* dataset: histogram of K. The histogram covers the 100 instances corresponding to each value of Δ_x.

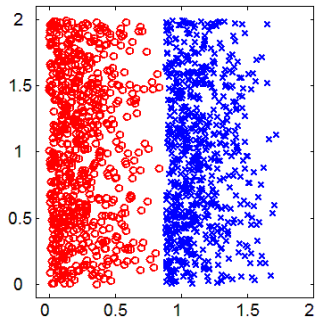
Moreover, examples of clustering solutions obtained by LSS-GCSS on different instances of the *2bars* dataset are shown in Figure 5.14.

Results

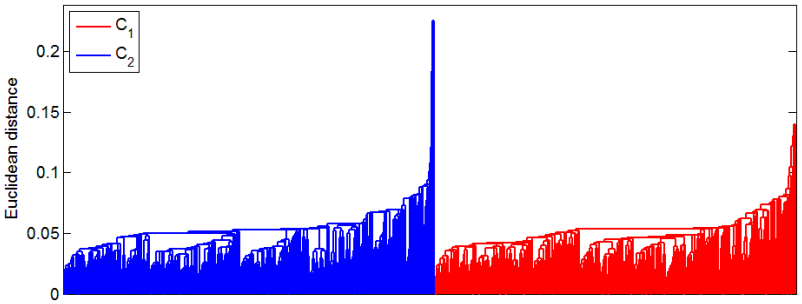
The clustering results illustrated in Figures 5.12 and 5.13 indicate that LSS-GCSS successfully deals with both touching and overlapping versions of the *2bars* dataset. LSS-GCSS begins to not distinguish between both clusters only when the frontier region disappears ($\Delta_x < -0.3$) and the overlapping degree becomes intense enough to lead to a single-cluster scenario (see Figure 5.14).

In addition, more specific conclusions can be drawn:

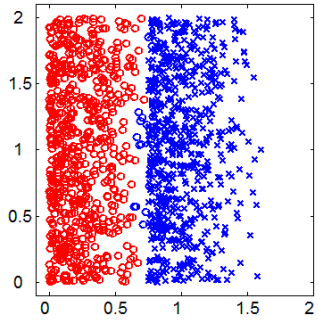
- The total CI averaged-value (considering all the instances along every value of Δ_x) equals 0.981, which indicates a really successful behaviour of LSS-GCSS in global terms. In addition, two clusters are identified ($K = 2$) in 97.42% of the instances.
- Extremely long shiftings of $\mathbf{C}_2$ ($\Delta_x \leq -0.6$) move the clustering scenario away from the expected configuration of the *2bars* dataset and get it closer to a single-cluster configuration more typical of the *1unif* dataset. Under such conditions, it is reasonable that clustering solutions with $K \neq 2$ begin to appear (see section 5.2.1).



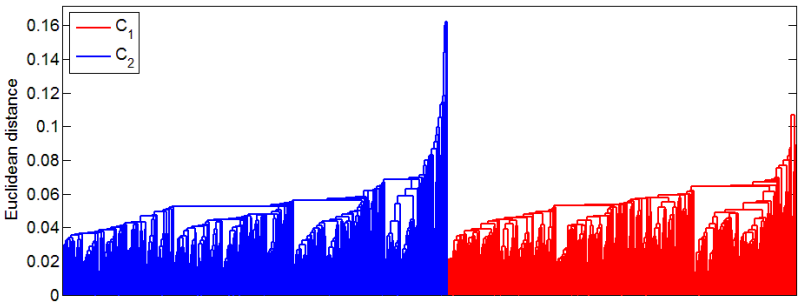
(a) *2bars* dataset: scatterplot.



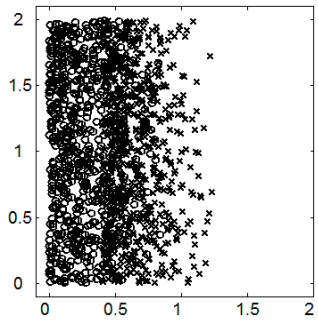
(b) *2bars* dataset: LSS-GCSS dendrograms ($\Delta_x = -0.12,\ K = 2,\ CI = 1$).



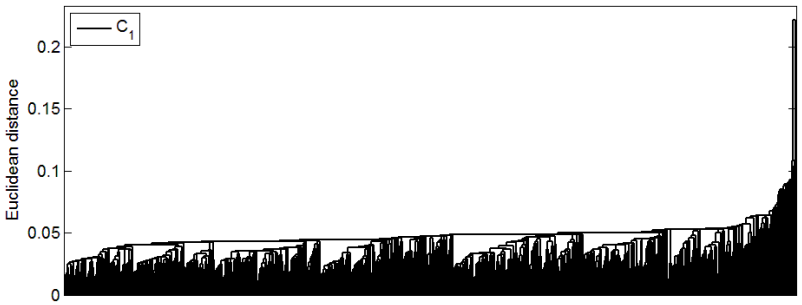
(c) *2bars* dataset: scatterplot.



(d) *2bars* dataset: LSS-GCSS dendrograms ($\Delta_x = -0.25,\ K = 2,\ CI = 0.984$).



(e) *2bars* dataset: scatterplot.



(f) *2bars* dataset: LSS-GCSS dendrogram ($\Delta_x = -0.6,\ K = 1,\ CI = 0.506$).

Figure 5.14: *2bars* dataset: clustering solutions by LSS-GCSS.

5.2.3 Unbalanced clusters

Input data

The performance of LSS-GCSS in the face of unbalanced clusters is studied in the present section. To that effect, the *2ubGauss* dataset is defined. It comprises two Gaussian distributed 2-dimensional unbalanced clusters ($D = 2,\ K_{opt} = 2$). The unbalancing between both clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) is ruled over by the unbalancing ratio R ($N_1 = 100,\ N_2 = R\,N_1,\ \forall\, R \in \mathbb{Z}^+ \mid R \geq 2$). Similarly to the *2Gauss* dataset (see section 5.2.2.2), both clusters are completely characterised by their respective means (μ_1 and μ_2) and standard deviations (σ_1 and σ_2). They both have unitary variance ($\sigma_1 = \sigma_2 = \sigma = 1$) and their degree of overlapping is ruled over by the distance between their means (d), which is again measured in units of σ ($d = n\,\sigma,\ \forall\, n \in \mathbb{R}^+$).

Characterisation

- The study covers multiple values of both the overlapping degree between clusters ($n \in [2, 10]$) and the balancing ratio ($R \in \{2, 10\}$).
- 100 instances of the *2ubGauss* dataset are tested per each combination of n and R.

Cluster analysis

Firstly, the performance of LSS-GCSS in terms of both CI and K is shown in Figures 5.15 and 5.16. Moreover, examples of clustering solutions obtained by LSS-GCSS on different instances of the *2ubGauss* dataset are shown in Figure 5.17.

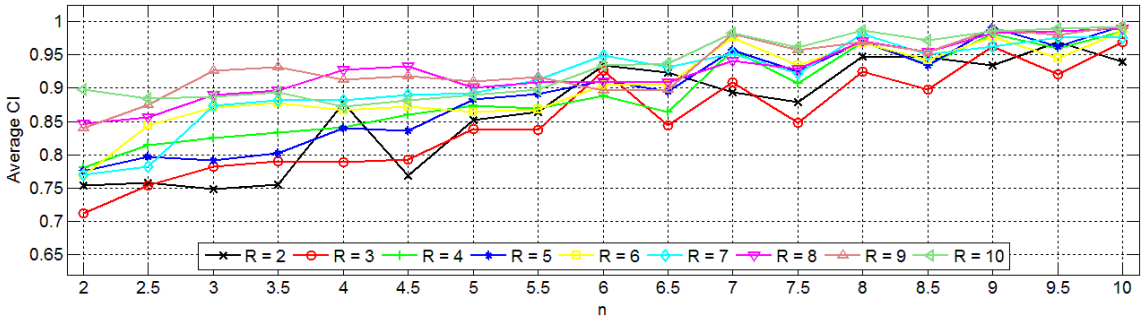


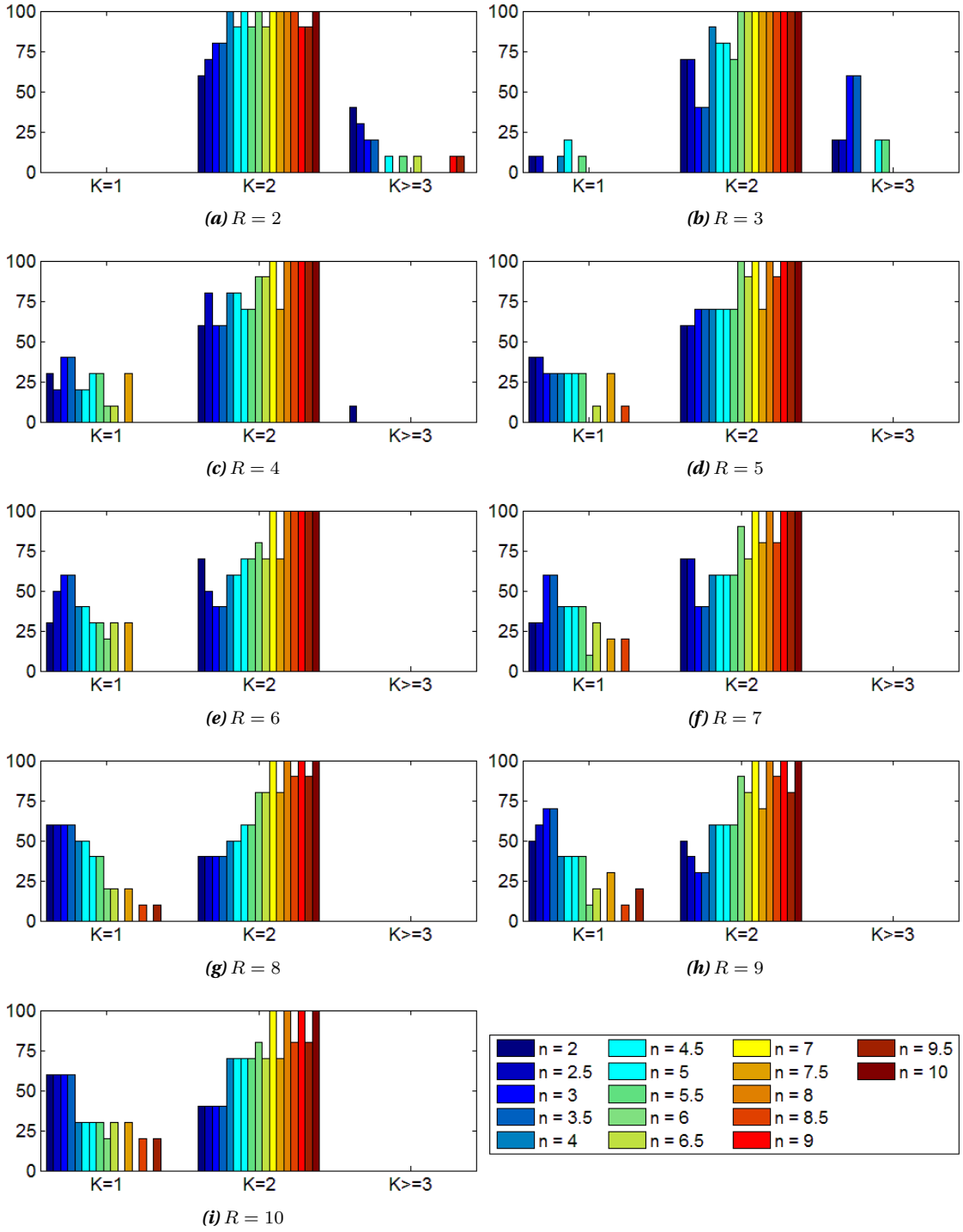
Figure 5.15: Performance of LSS-GCSS on *2ubGauss* dataset. CI values averaged along 100 instances.

Results

In general terms, the evaluation of the clustering results shown in Figures 5.15 and 5.16 indicates that LSS-GCSS tends to successfully identify unbalanced clusters to the extent that the degree of separation/overlapping between clusters (n) grows with the unbalancing ratio (R).

In addition, more specific conclusions can be drawn:

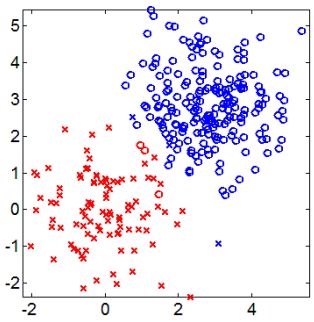
- On the one hand, a proper understanding of the CI averaged-values shown in Figure 5.15 involves noticing that the higher the balancing ratio, the higher the values of CI, regardless of the quality of the clustering solution. Let the instance illustrated in Figures 5.17g and 5.17h be considered by way of example: only one cluster ($K = 1$) is identified by LSS-GCSS, but an apparently high value of CI is reached ($CI = 0.909$), since $R = 10$ and only a 9.01% $\left((\%) = 100\frac{1}{R+1}\right)$ of the objects are misassigned in the obtained clustering solution.
- On the other hand, similarly to the *2Gauss* dataset, the fact that clusters are overlapped causes the maximum value of CI ($CI = 1$) not to be reached although certainly accurate results are obtained ($K = 2$), since there will always be misassigned objects as a result of the overlapping (see Figures 5.17a and 5.17b).



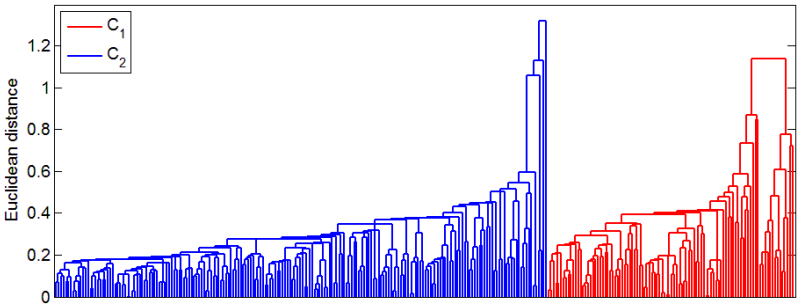
(a) $R = 2$ ***(b)*** $R = 3$ ***(c)*** $R = 4$ ***(d)*** $R = 5$ ***(e)*** $R = 6$ ***(f)*** $R = 7$ ***(g)*** $R = 8$ ***(h)*** $R = 9$ ***(i)*** $R = 10$

Figure 5.16: Performance of LSS-GCSS on *2ubGauss* dataset: histograms of K. Every histogram corresponds to a specific value of R and includes the 100 instances corresponding to each value of n.

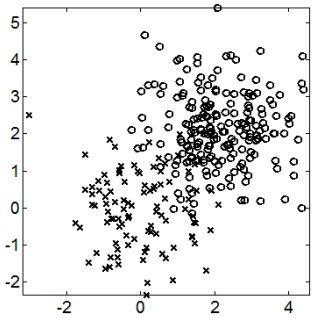
- The crux of the matter is the relationship between the unbalancing ratio (R) and the degree of separation/overlapping between clusters (n). LSS-GCSS proves to be able to handle unbalancing inasmuch as unbalanced clusters are not highly overlapped. Figure 5.16 clearly shows that, if the unbalancing ratio grows, LSS-GCSS needs of higher degrees of separation between unbalanced clusters in order to be able to identify two clusters in the dataset.



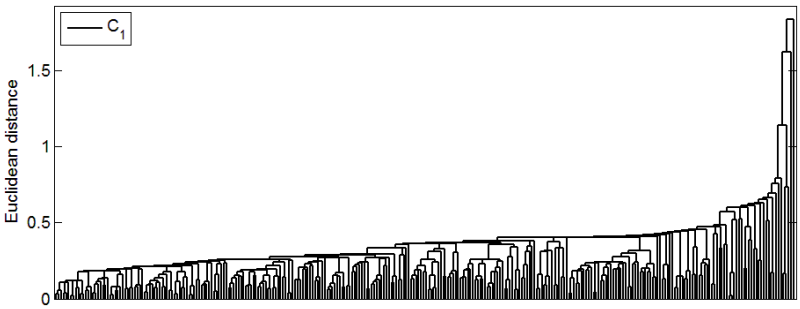
(a) *2ubGauss* dataset: scatterplot.



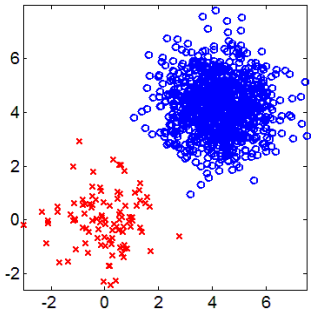
(b) *2ubGauss* dataset: LSS-GCSS dendrograms ($R = 2,\ n = 4,\ K = 2,\ CI = 0.98$).



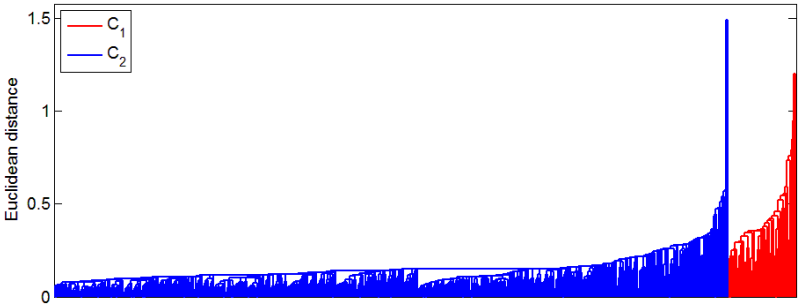
(c) *2ubGauss* dataset: scatterplot.



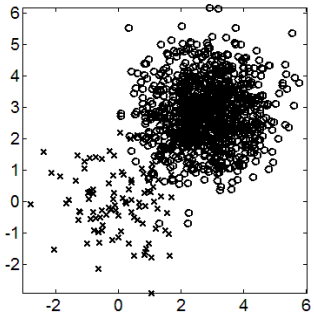
(d) *2ubGauss* dataset: LSS-GCSS dendrogram ($R = 2,\ n = 3,\ K = 1,\ CI = 0.667$).



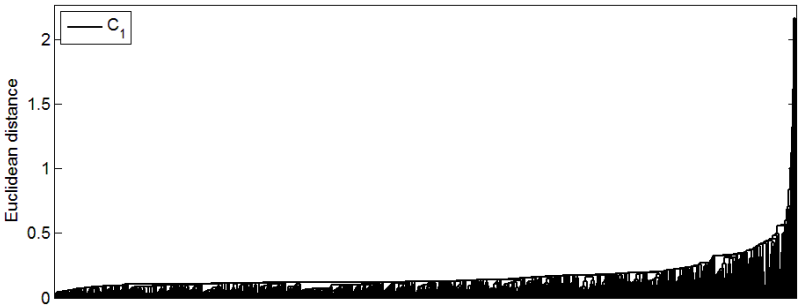
(e) *2ubGauss* dataset: scatterplot.



(f) *2ubGauss* dataset: LSS-GCSS dendrograms ($R = 10,\ n = 6,\ K = 2,\ CI = 1$).



(g) *2ubGauss* dataset: scatterplot.



(h) *2ubGauss* dataset: LSS-GCSS dendrogram ($R = 10,\ n = 4,\ K = 1,\ CI = 0.909$).

Figure 5.17: *2ubGauss* dataset: clustering solutions by LSS-GCSS.

- It has also been verified that the effect of dimensionality does not affect to the present conclusions and considerations: if data dimensionality grows ($D > 2$), the distance between clusters has to grow in absolute terms in order for the degree of separation/overlapping between clusters to remain constant and, therefore, LSS-GCSS still successfully deals with the unbalanced clusters present in the dataset.

- Equivalent clustering results are reached and same conclusions are drawn (LSS-GCSS deals properly with the unbalancing as it grows with the separation between clusters) in the case of two uniformly distributed unbalanced clusters.

5.2.4 Multiple-cluster datasets

Input data

The performance of LSS-GCSS in the face of multiple-cluster datasets is studied in the present section. To that effect, the *Munif* dataset is defined. It comprises M uniformly-distributed 2-dimensional balanced clusters ($K_{opt} = M,\ D = 2,\ N_i = 100,\ N = 100\,M$). Assuming that the ith object in the dataset ($\mathbf{x}_i$) is centred around the coordinates (x_{i_0}, y_{i_0}), all clusters have the same variance ($x_{i_1} \in \left[x_{i_0} - \frac{1}{2}, x_{i_0} + \frac{1}{2}\right],\ x_{i_2} \in \left[y_{i_0} - \frac{1}{2}, y_{i_0} + \frac{1}{2}\right],\ \forall\, \mathbf{x}_i$). Similarly to the 2unif dataset (see section 5.2.2.1), every cluster is separated a distance d from their neighbours, so that clusters are moved along the two dimensions of the dataset (see Figure 5.20a for further details).

Characterisation

- The study covers multiple values of both the distance between clusters ($d \in [0, 0.8]$) and the total number of clusters in the dataset ($M \in \{3, 100\}$).
- 100 instances of the *Munif* dataset are tested per each combination of parameters d and M.

Cluster analysis

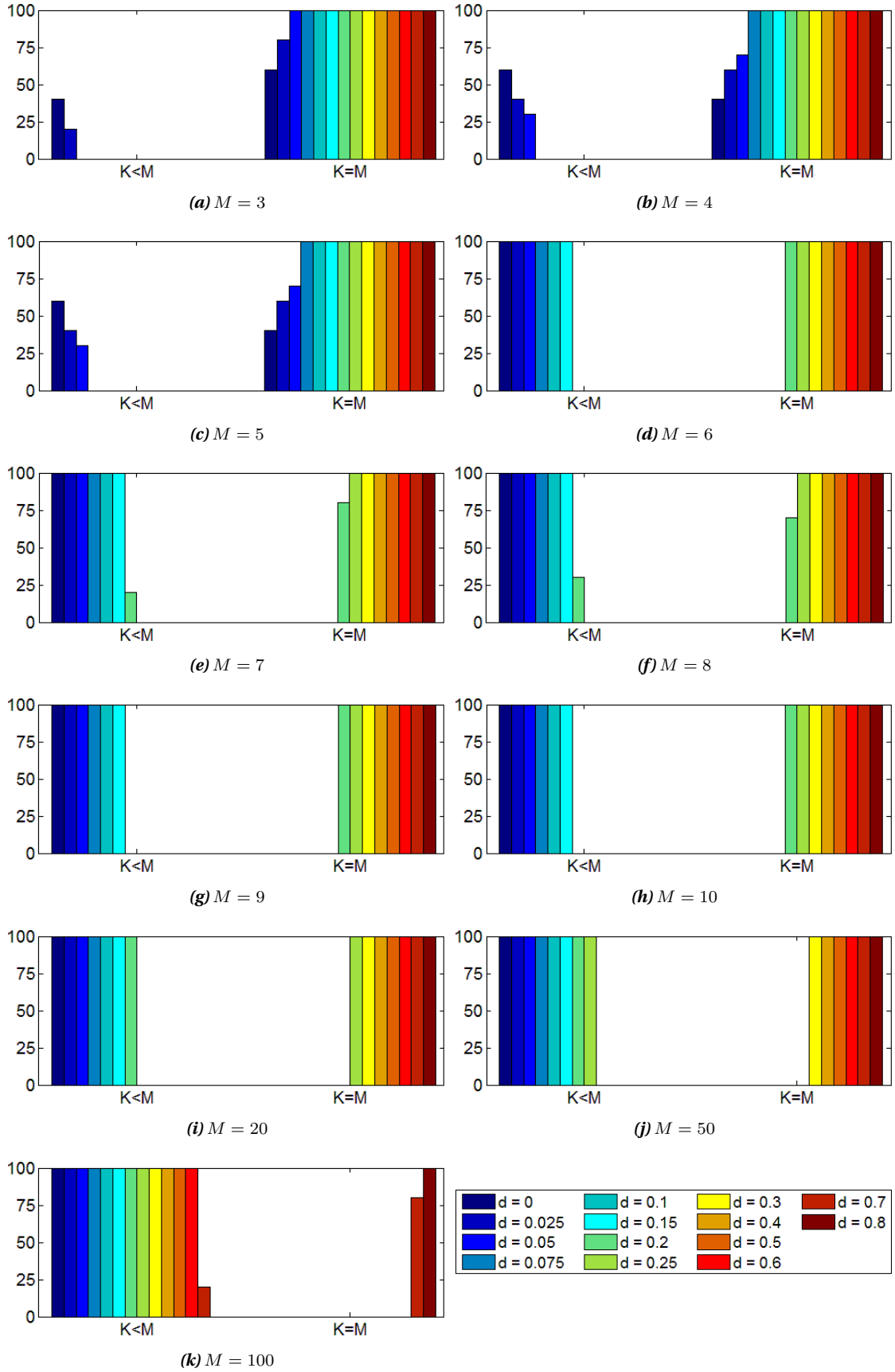
Firstly, the performance of LSS-GCSS in terms of both K and CI is shown in Figures 5.18 and 5.19. Moreover, the clustering solution obtained by LSS-GCSS in the face of an instance of the *Munif* dataset that contains 32 clustes ($M = 32$) is shown in Figure 5.20 by way of example.

Results

In general terms, the evaluation of the clustering results shown in Figures 5.19 and 5.18 indicates that LSS-GCSS is able to successfully identify large amounts of clusters in a dataset. Nonetheless, accurate clustering results are subject to the fact that the separation between clusters (d) grows with the true number of clusters in the ground truth (M).

In addition, more specific conclusions can be drawn:

- Since the merging of small candidate clusters ($N_i < 0.01\,N$) is not required to account for the merging criteria in the agglomeration process performed by LSS-GCSS, datasets that com-



(a) $M = 3$ ***(b)*** $M = 4$

(c) $M = 5$ ***(d)*** $M = 6$

(e) $M = 7$ ***(f)*** $M = 8$

(g) $M = 9$ ***(h)*** $M = 10$

(i) $M = 20$ ***(j)*** $M = 50$

(k) $M = 100$

Figure 5.18: Performance of LSS-GCSS on *Munif* dataset: histograms of K. Every histogram corresponds to a specific value of M and includes the 100 instances corresponding to each value of d.

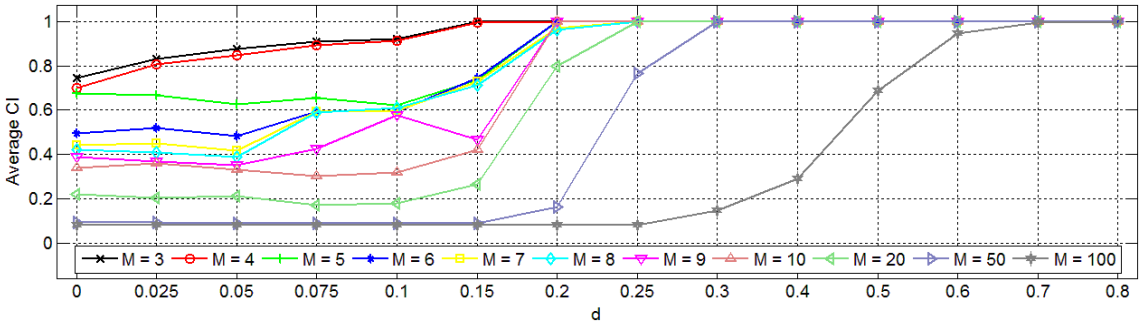
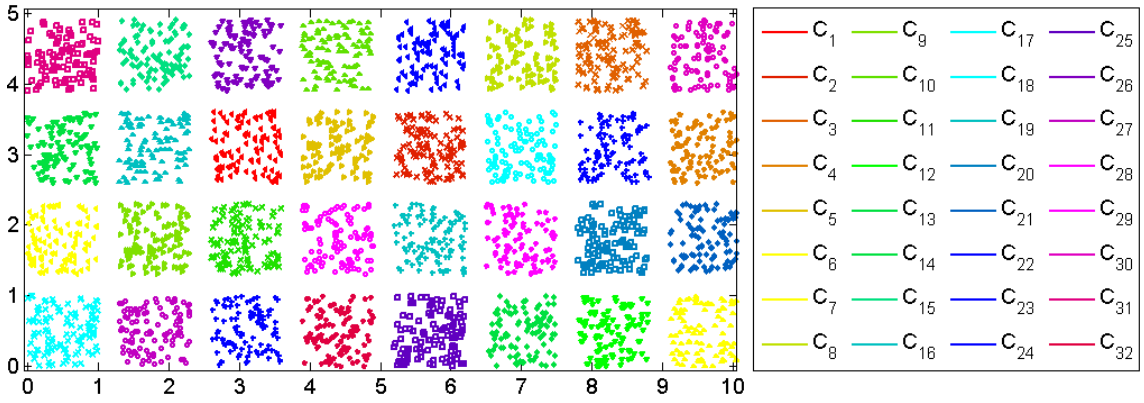
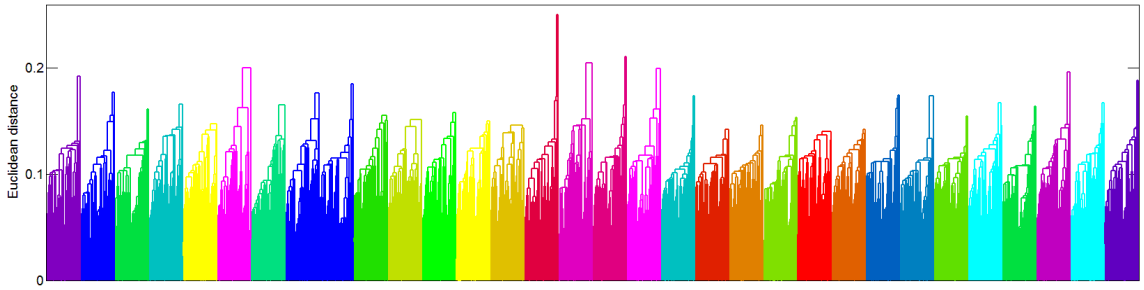


Figure 5.19: Performance of LSS-GCSS on *Munif* dataset. CI values averaged along 100 instances.



(a) *Munif* dataset: scatterplot ($M = 32$, $d = 0.3$).



(b) *Munif* dataset: LSS-GCSS dendrograms ($K = 32$, $CI = 1$).

Figure 5.20: *Munif* dataset: clustering solution by LSS-GCSS.

prise more than a hundred of balanced clusters ($M > 100$) cannot be properly analysed with LSS-GCSS (see line 7 of Algorithm 5 in section 4.2). In case of clusters being unbalanced, LSS-GCSS could be able to identify them, regardless of the value of M, when the size of at least one of them is large enough ($N_i < 0.01\,N < N_j$), but never between two small enough clusters ($N_i, N_j < 0.01\,N$).

Nonetheless, such a scenario is certainly unlikely in practice, since the goal of cluster analysis is to provide simplified and understandable descriptions of the underlying structure of the data; a clustering solution that comprises a large amount of clusters may easily fail in being understandable and can complicate the interpretation of the results (see sections 1.2.3 and 1.2.3.2 for further details). Moreover, if it is necessary, there exist data mining strategies that allow clustering algorithms to easily overcome limitations like this one (see Chapter 6 for further details).

- The crux of the matter is again the relationship between the number of clusters (M) and their degree of separation (n). LSS-GCSS proves to be able to handle multiple clusters inasmuch as they are separated enough. Figure 5.18 clearly shows that, if the amount of clusters grows, LSS-GCSS needs of higher distances between them in order to be able to accurately estimate the real number of clusters present in the dataset.

- It has also been verified that the effect of dimensionality does not affect to the present conclusions: if data dimensionality grows $(D > 2)$, the distance between clusters has to grow in absolute terms in order for the separation between clusters to remain constant and, therefore, LSS-GCSS still successfully handles the large amount of clusters present in the dataset.

- Equivalent clustering results are reached and same conclusions are drawn in the case of M Gaussian distributed clusters.

5.2.5 Concentric clusters

Scenarios with concentric clusters are typically troublesome for centre-based HPC algorithms (*e.g.* k-means), which are unable to deal with clusters of this kind, imposing globular-shaped clusters on the data (Fred and Leitão, 2003). Thus, the performance of LSS-GCSS algorithm in the face of datasets that present touching and overlapped clusters is studied in the present section, which includes datasets that comprise both circular-shaped clusters (see section 5.2.5.1) and spiral-shaped clusters (see section 5.2.5.2).

5.2.5.1 The *3rings* dataset

Input data

The *3rings* dataset is composed of fifteen hundred objects $(N = 1500)$ structured into three 2-dimensional uniformly-distributed ring-shaped concentric clusters $(D = 2,\ K_{opt} = 3)$. All three clusters $(\mathbf{C}_1,\ \mathbf{C}_2$, and $\mathbf{C}_3)$ are perfectly balanced $(N_1 = N_2 = N_3 = 500)$ and separated the same distance d. According to polar coordinates, the radius of the ith object in the dataset $(\mathbf{x}_i)$ is denoted as r_i and its value fences in a specific range, depending on which cluster the object belongs to: $r_i \in [0, 0.75]$, $\forall\, \mathbf{x}_i \in \mathbf{C}_1$; $r_i \in [0.75+d, 1.25+d]$, $\forall\, \mathbf{x}_i \in \mathbf{C}_2$; $r_i \in [1.25+2d, 1.75+2d]$, $\forall\, \mathbf{x}_i \in \mathbf{C}_3$.

Characterisation

- The study covers multiple values of the distance between clusters $(d \in [0.05, 0.5])$.

- 100 instances of the *3rings* dataset are tested per each value of parameter d.

Cluster analysis

Figures 5.21 and 5.22 show the performance of LSS-GCSS in terms of both CI and K.

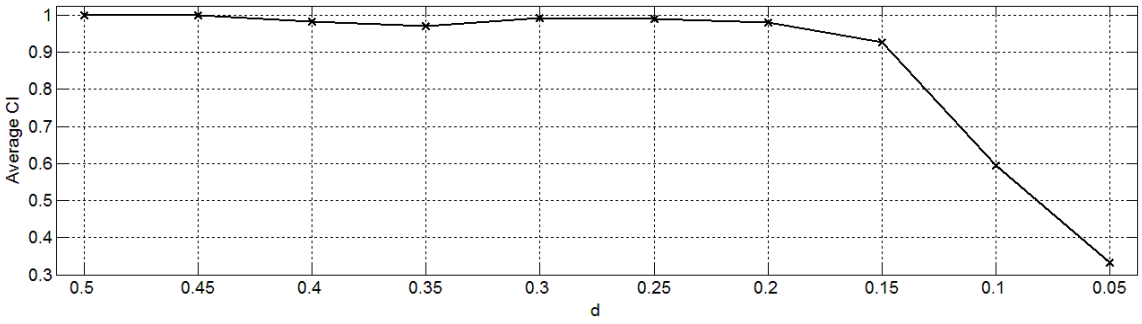


Figure 5.21: Performance of LSS-GCSS on *3rings* dataset. CI values averaged along 100 instances.

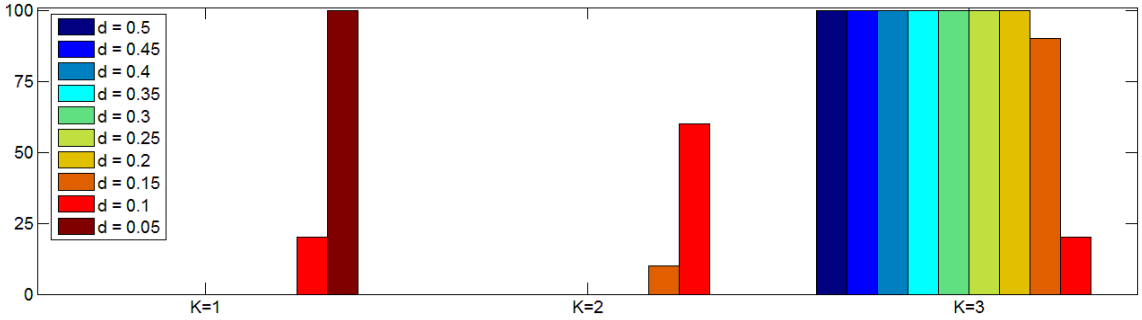


Figure 5.22: Performance of LSS-GCSS on *3rings* dataset: histogram of K. The histogram covers the 100 instances corresponding to each value of d.

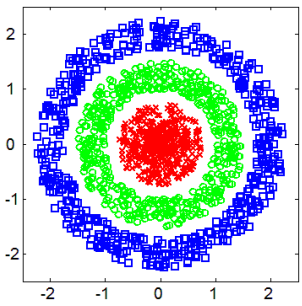
Moreover, examples of clustering solutions obtained by LSS-GCSS on different instances of the *3rings* dataset are shown in Figure 5.23.

Results

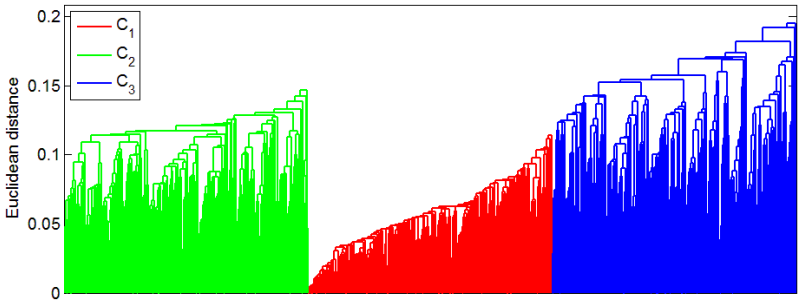
The clustering results illustrated in Figures 5.21 and 5.22 indicate that LSS-GCSS is able to successfully identify ring-shaped concentric clusters. LSS-GCSS begins to not identify concentric clusters properly only when the frontier region between rings tends to disappear ($d < 0.15$) and a single-cluster scenario is progressively reached (see Figure 5.23).

In addition, more specific conclusions can be drawn:

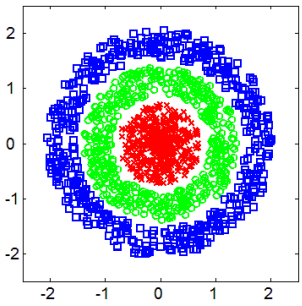
- The total CI averaged-value considering all the instances with $d \geq 0.15$ equals 0.98, which indicates a really successful behaviour of LSS-GCSS in global terms. In addition, the three rings are successfully identified ($K = 3$) in 98.75% of same set of instances with $d \geq 0.15$.
- Extremely low separations between rings ($d \leq 0.05$) move the clustering scenario away from the expected configuration of the *3rings* dataset and get it closer to a single-cluster configuration more typical of a concentric-shaped *1unif* dataset. Under such conditions, it is reasonable that clustering solutions tend to identify one single cluster in the dataset ($K = 1$).



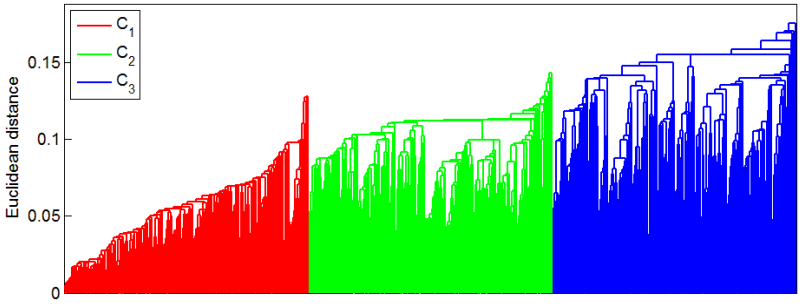
(a) *3rings* dataset: scatterplot.



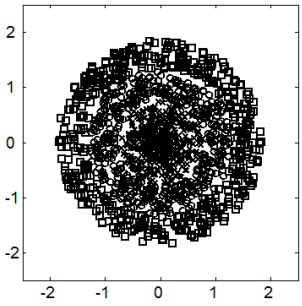
(b) *3rings* dataset: LSS-GCSS dendrograms ($d = 0.25,\ K = 3,\ CI = 1$).



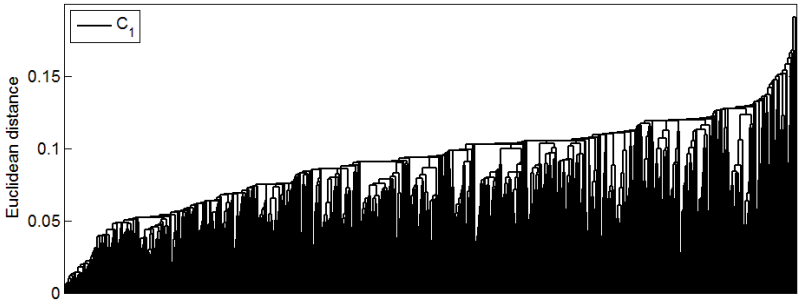
(c) *3rings* dataset: scatterplot.



(d) *3rings* dataset: LSS-GCSS dendrograms ($d = 0.15,\ K = 3,\ CI = 1$).



(e) *3rings* dataset: scatterplot.



(f) *3rings* dataset: LSS-GCSS dendrogram ($d = 0.05,\ K = 1,\ CI = 0.333$).

Figure 5.23: *3rings* dataset: clustering solutions by LSS-GCSS.

5.2.5.2 The *2spirals* dataset

Input data

The *2spirals* dataset comprises a slightly random amount ($N \in \{1025, 1175\}$) of 2-dimensional objects ($D = 2$) grouped into two spiral-shaped concentric clusters ($K_{opt} = 2$). Both clusters ($\mathbf{C}_1$ and $\mathbf{C}_2$) are unbalanced ($N_1 = 600,\ N_2 \in \{425, 575\}$).

Identifying the 2-dimensional space of the dataset as the complex plane, every cluster describes the trajectory of an arithmetic spiral, whose ith point is defined as:

$$r_i = \alpha_i \, e^{j\theta_i} + \delta_i \tag{5.1}$$

being α_i, θ_i and δ_i defined in the *2spirals* dataset as follows:

$$\alpha_i = \frac{i-1}{200L} \quad , \quad \theta_i = \frac{\pi}{100}(i-1) \quad , \quad \delta_i = 0.01 \tag{5.2}$$

where L is the number of loops the spiral traces –starting from its centre with zero phase– and it has been set at the value of 3.

Thus, the ith object in the dataset $(\mathbf{x}_i)$, which belongs to the jth cluster $(\mathbf{C}_j)$, is defined as:

$$\mathbf{x}_i = e^{j\theta_0^{(j)}} r_i + \eta \tag{5.3}$$

where the term η is a Gaussian noise of zero mean and variance σ_η^2 $\left(\sigma_\eta^2 \gtrapprox 10^{-4}\right)$ and $\theta_0^{(j)}$ is the initial phase of the spiral traced by objects belonging to $\mathbf{C}_j$. In the *2spirals* dataset, $\theta_0^{(1)} = 0$ and $\theta_0^{(2)} = \theta_0$. Hence, the parameter θ_0 allows to define the separation between both spirals; *i.e.* the separation between $\mathbf{C}_1$ and $\mathbf{C}_2$ increases inasmuch as the value of θ_0 moves away from 0 and towards π.

Finally, it is worth noting that the unbalancing ratio between both clusters is also ruled over by θ_0, since, depending on its value, the second spiral traces more or less loops and, therefore, $\mathbf{C}_2$ comprises more or less objects.

Characterisation

- The study covers multiple values of the distance between clusters $\left(\theta_0 \in \left[\frac{\pi}{4}, \frac{7\pi}{4}\right]\right)$.
- 100 instances of the *2spirals* dataset are tested per each value of parameter θ_0.

Cluster analysis

Figure 5.24 shows the performance of LSS-GCSS in terms of CI.

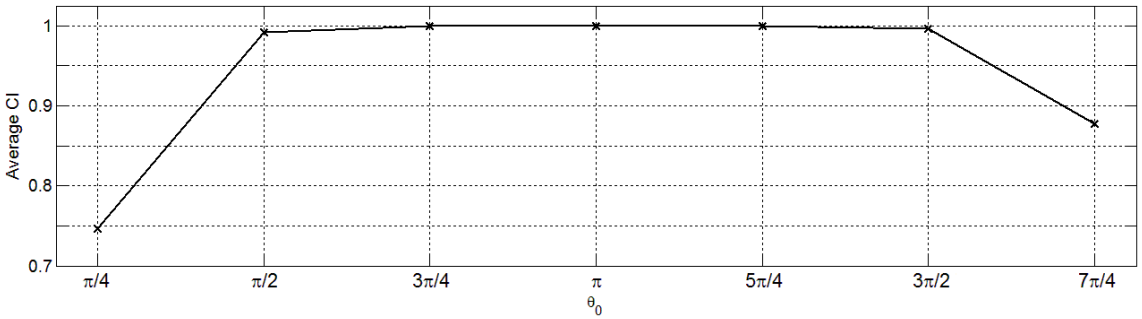
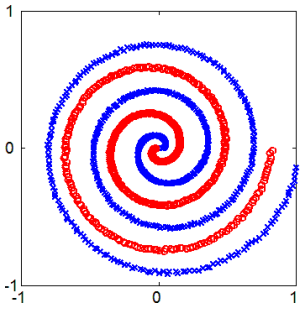


Figure 5.24: Performance of LSS-GCSS on *2spirals* dataset. CI values averaged along 100 instances.

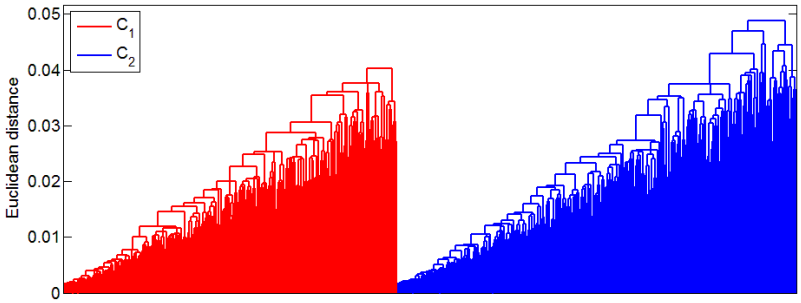
Moreover, examples of clustering solutions obtained by LSS-GCSS on different instances of the *2spirals* dataset are shown in Figure 5.25.

Results

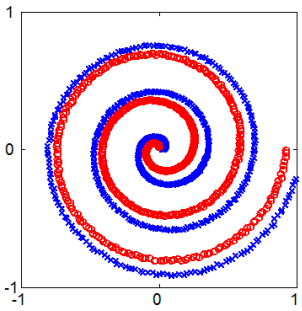
The clustering results illustrated in Figure 5.24 indicates that LSS-GCSS is able to successfully identify spiral-shaped concentric clusters. Again, LSS-GCSS begins to not identify clusters properly only when the frontier region tends to disappear $\left(-\frac{\pi}{4} \leq \theta_0 \leq \frac{\pi}{4}\right)$ (see Figure 5.25).



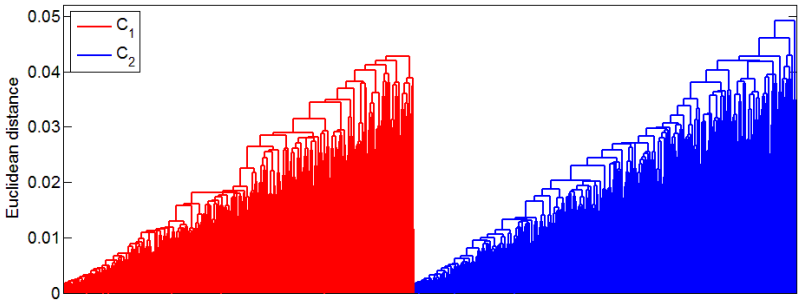
(a) *2spirals* dataset: scatterplot.



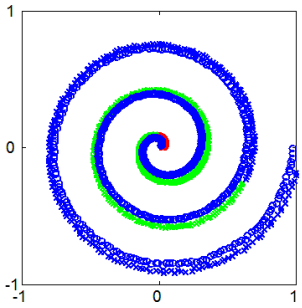
(b) *2spirals* dataset: LSS-GCSS dendrograms ($\alpha = \pi$, $K = 2$, $CI = 1$).



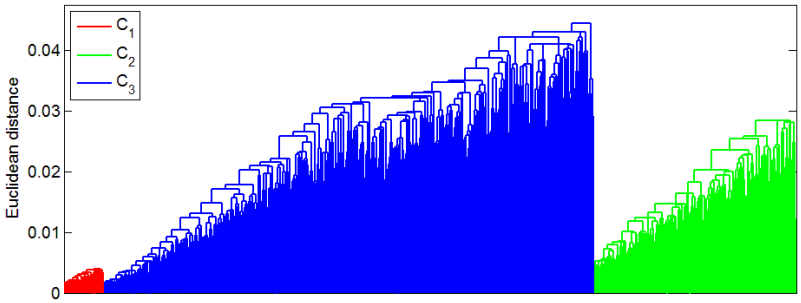
(c) *2spirals* dataset: scatterplot.



(d) *2spirals* dataset: LSS-GCSS dendrograms ($\alpha = \frac{\pi}{2}$, $K = 2$, $CI = 1$).



(e) *2spirals* dataset: scatterplot.



(f) *2spirals* dataset: LSS-GCSS dendrograms ($\alpha = \frac{\pi}{4}$, $K = 3$, $CI = 0.766$).

Figure 5.25: *2spirals* dataset: clustering solutions by LSS-GCSS.

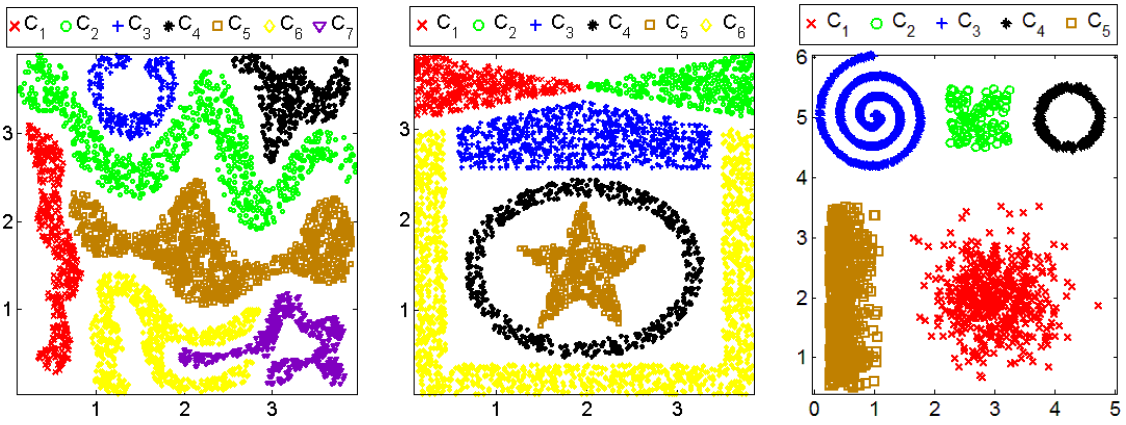
The total CI averaged-value (considering all the instances along every value of parameter θ_0) equals 0.946, which indicates a really successful behaviour of LSS-GCSS algorithm in global terms. In addition, the two spirals are successfully identified ($K = 2$) in 96.87% of the instances of the *2spirals* dataset.

5.2.6 Arbitrary-shaped clusters

Input data

Scenarios with differently-shaped clusters can be certainly troublesome for many clustering algorithms, since they may impose determinate shapes to the clusters they identify, which can make them unable to deal with the real shape of the clusters actually present in the dataset. Thus, the performance of LSS-GCSS algorithm in the face of clusters with different shapes is studied in the

present section, which includes three distinct datasets that comprise different mixes of arbitrary-shaped clusters (see Figure 5.26):



(a) The *Mix#1* dataset. ***(b)*** The *Mix#2* dataset. ***(c)*** The *Mix#3* dataset.

Figure 5.26: Datasets with a mix of arbitrary-shaped clusters.

- The *Mix#1* dataset (see Figure 5.26a)) comprises a composition of approximately four thousand two-dimensional objects uniformly-distributed into seven irregularly-shaped clusters ($N = 3958$, $D = 2$, $K_{opt} = 7$) of different sizes ($N_1 = 385$, $N_2 = 793$, $N_3 = 247$, $N_4 = 386$, $N_5 = 1002$, $N_6 = 729$, $N_7 = 416$).

- The *Mix#2* dataset (see Figure 5.26b)) comprises a composition of approximately four thousand two-dimensional objects uniformly-distributed into six clusters ($N = 3960$, $D = 2$, $K_{opt} = 6$) of different sizes ($N_1 = 354$, $N_2 = 350$, $N_3 = 839$, $N_4 = 586$, $N_5 = 386$, $N_6 = 1445$), which adopt different geometric shapes (triangles, rectangles, circles, stars, etc.).

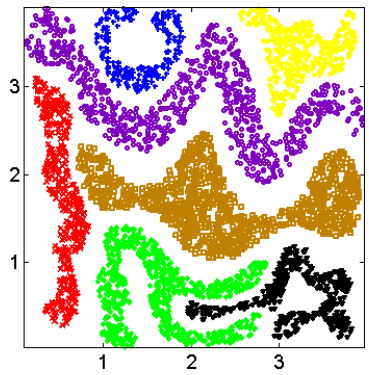
- The *Mix#3* dataset (see Figure 5.26c)) comprises a composition of approximately two thousand two-dimensional objects structured into five clusters ($N = 1916$, $D = 2$, $K_{opt} = 5$) of different sizes ($N_1 = 500$, $N_2 = 100$, $N_3 = 600$, $N_4 = 200$, $N_5 = 516$) and distributions (Gaussian, uniform, spiral, ring and bar).

Cluster analysis

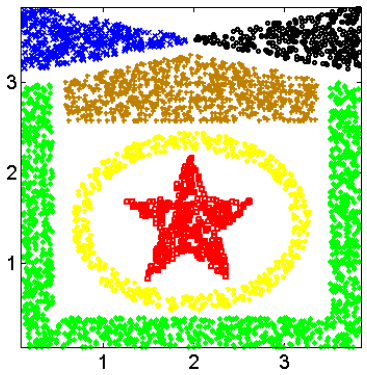
Figure 5.27 shows the clustering solutions obtained by LSS-GCSS in the face of *Mix#1*, *Mix#2* and *Mix#3* datasets.

Results

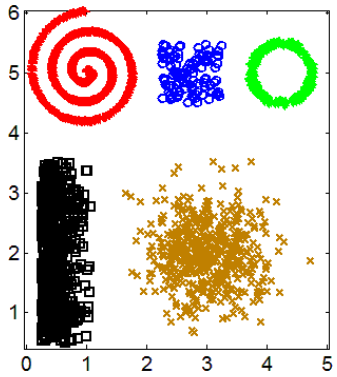
The clustering results illustrated in Figure 5.27 indicates that LSS-GCSS successfully deals with arbitrary-shaped clusters, since it matches the ground truth solution 100% ($CI = 1$) on the three proposed datasets.



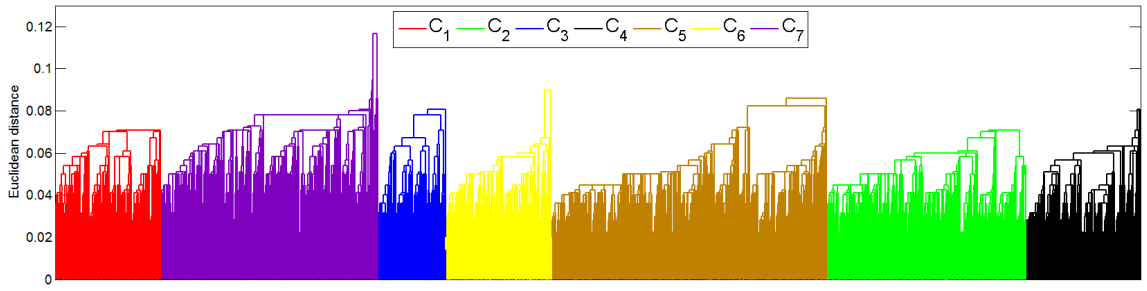
(a) *Mix#1* dataset: scatterplot.



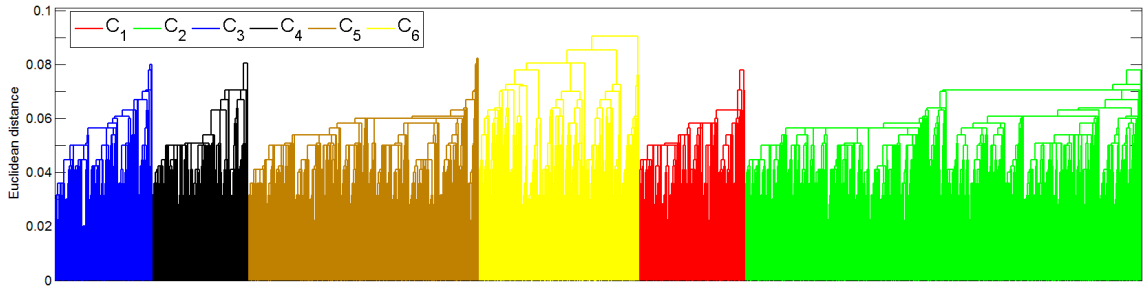
(b) *Mix#2* dataset: scatterplot.



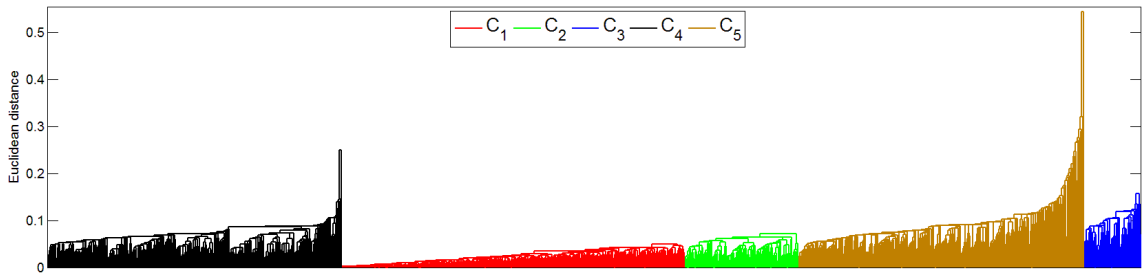
(c) *Mix#3* dataset: scatterplot.



(d) *Mix#1* dataset: LSS-GCSS dendrograms.



(e) *Mix#2* dataset: LSS-GCSS dendrograms.



(f) *Mix#3* dataset: LSS-GCSS dendrograms.

Figure 5.27: Datasets with a mix of arbitrary-shaped clusters: clustering solutions by LSS-GCSS.

It is worth noting that neither the unbalancing between clusters nor the presence of multiple clusters lead to troublesome clustering situations (see sections 5.2.3 and 5.2.4 for further details) in none of the proposed datasets, since, although frontier regions can become certainly narrow and lengthy –specially in *Mix#1* and *Mix#2* datasets–, clusters are not overlapped each other.

5.3 Real datasets

The main goal of the present section is to study the performance of the LSS-GCSS in the face of different real-word clustering problems. To that effect, a diversity of real datasets that includes both clusters of different characteristics (dimensionality, unbalancing ratio, degree of overlapping, number of clusters, etc.) and data from different nature and origin (speech, images, text, etc.) is selected. Every of the following datasets is used as benchmark in the literature (with the aim of performing tests and comparisons under equal and known conditions) and is of public access from the Machine Learning Repository of the University of California, Irvine (UCI)[1]:

- The *Wine* dataset (see section 5.3.1) is a well-known benchmark dataset in the machine learning field.
- The *Iris* dataset (see section 5.3.2) is probably the most widely used benchmark dataset in pattern recognition literature.
- The *WDBC* datasets (see section 5.3.3) include data from medical images and are benchmark datasets also highly used in classification contexts.
- The *SAD* dataset (see section 5.3.4) includes preprocessed speech data and is a well-known benchmark dataset in the speech processing field.
- The *MiniNews* dataset (see section 5.3.5) includes text data and is also a widely used benchmark dataset in the information retrieval field.

Furthermore, all the datasets selected in this section present real numerical features and the proximity between objects is measured by means of either the Euclidean distance (see equation 2.8) or the Cosine distance (see equation 2.10), depending on the nature of the dataset.

Finally, aside from comparing the number of identified clusters (K) with the real number of clusters in the ground truth solution (K_{opt}), every twofold clustering solution obtained by LSS-GCSS in the present section is evaluated under several validation methods:

- **External validation**
  - The Consistency Index (CI), which provides external validation for the partitional clustering solution (see section 2.4.1 for further details).
- **Internal validation**
  - The Silhouette Coefficient $(\overline{S})$, which provides internal validation for the partitional clustering solution (see section 2.4.2 for further details).

[1] http://archive.ics.uci.edu/ml/index.html

- The Kruskal-Wallis statistical test, which measures the presence or absence of significant differences among the populations of the identified clusters in the partitional clustering solution (see section 2.4.3 for further details).
- The Cophenetic Correlation Coefficient ($CPCC$), which provides internal validation for the hierarchical clustering solution corresponding to every identified cluster (see section 2.4.4 for further details).

5.3.1 The *Wine* dataset

Input data

The *Wine* dataset[2] results of a chemical analysis of one hundred and seventy-eight different wines ($N = 178$) grown in the same region in Italy, but derived from three distinct cultivars ($K_{opt} = 3$). The analysis determine the quantities of thirteen constituents ($D = 13$) found in each of the three types of wines (Forina et al., 1990).

In classification contexts, the *Wine* dataset is considered to give rise to a not very challenging classification problem with well-behaved class structures.

Characterisation

- The dataset is slightly unbalanced ($N_1 = 59,\ N_2 = 71,\ N_3 = 48$).
- The objects in this dataset are characterised by the 13 features next listed, along with their range of values:
  - Alcohol ($[11.03, 14.83]$)
  - Malic acid ($[0.74, 5.8]$)
  - Ash ($[1.36, 3.23]$)
  - Alcalinity of ash ($[10.6, 30]$)
  - Magnesium ($[70, 162]$)
  - Total phenols ($[0.98, 3.88]$)
  - Flavanoids ($[0.34, 5.08]$)
  - Nonflavanoid phenols ($[0.13, 0.66]$)
  - Proanthocyanins ($[0.41, 3.58]$)
  - Color intensity ($[1.28, 13]$)
  - Hue ($[0.48, 1.71]$)
  - OD280/OD315 of diluted wines ($[1.27, 4]$)
  - Proline ($[278, 1680]$)
- All features are normalised to zero mean and unit variance prior to cluster analysis.
- The squared Euclidean distance (the square of the Euclidean distance) is used as proximity measure.

[2] http://archive.ics.uci.edu/ml/datasets/Wine

Cluster analysis

As shown in Figure 5.28, three different clusters are identified by the LSS-GCSS algorithm in the *Wine* dataset:

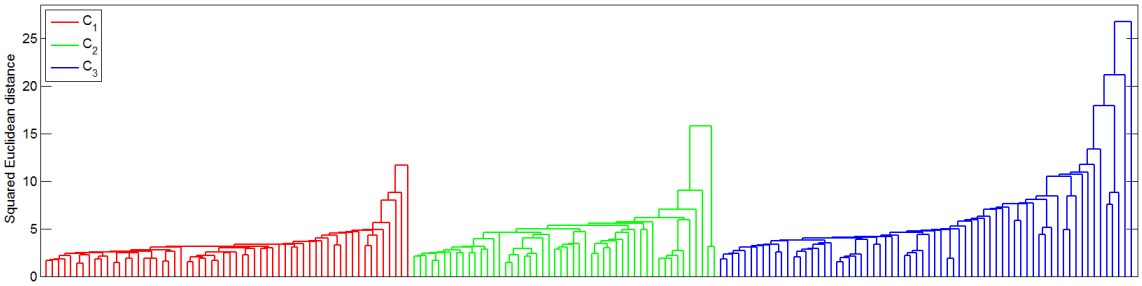


Figure 5.28: *Wine* dataset: clustering solution by LSS-GCSS.

Results

The evaluation of the obtained clustering results (see Tables 5.1 and 5.2) indicates that LSS-GCSS successfully identifies the three distinct cultivars present in the *Wine* dataset:

CI	$\overline{S}$	$CPCC$		
		$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$
0.938	0.392	0.808	0.624	0.746

Table 5.1: Validity measures for the clustering solution shown in Figure 5.28.

	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$
Cultivar 1	56	0	3
Cultivar 2	4	3	64
Cultivar 3	0	47	1

Table 5.2: Matching matrix of the clustering solution shown in Figure 5.28.

- The obtained clustering solution matches the ground truth in a high degree ($CI = 0.938$), which points out a proper identification of the real clusters present in the dataset.
- The matching matrix resulting from the calculation of CI illustrates the mapping between the different cultivars present in the dataset and the obtained clusters, as well as the misassignments performed by the clustering solution (11 misassigned objects out of 178).
- The value of the Silhouette Coefficient ($\overline{S} = 0.392$) indicates the presence of relatively low-compacted and/or low-isolated clusters, which assists in a better characterisation of the results. The value of $\overline{S}$ (whose behaviour is illustrated in Figure 2.3) needs to be carefully interpreted: despite being far from its maximum, it is positive and significantly higher than 0, which, at least, allows to dismiss the presence of incoherent clustering results.
- Finally, the $CPCC$ values confirms that the obtained dendrograms properly represent the structures and relationships between objects within their respective clusters. Moreover, the shape of dendrograms shown in Figure 5.28 suggest the presence of non-uniform clusters, with low-density regions in the frontiers between clusters, which would explain the few misassignments detected in the matching matrix.

In addition, the Kruskal-Wallis statistical hypothesis test indicates that there are significant differences ($p < 0.01$) between objects belonging to different clusters considering all pairs of clusters along every feature in the dataset (*i.e.* objects belonging to different clusters actually come from different statistical distributions), except for the three cases shown in Table 5.3.

Feature	p-value
Ash ($\mathbf{C}_1 - \mathbf{C}_2$)	0.536
Magnesium ($\mathbf{C}_2 - \mathbf{C}_3$)	0.011
Hue ($\mathbf{C}_1 - \mathbf{C}_3$)	0.368

Table 5.3: Non-significant differences in the clustering solution shown in Figure 5.28.

This results confirm the internal quality of the obtained clustering solution, since, aside from matching the ground truth in a high degree, the identified clusters present differences among them non-attributable to randomness; *i.e.* objects belonging to different clusters are actually not alike.

5.3.2 The *Iris* dataset

Input data

The *Iris* dataset[3] is one of the best known and most widely used databases in the pattern recognition field. It contains data about one hundred and fifty specimens ($N = 150$) belonging to three different types of iris plants ($K_{opt} = 3$): *Setosa*, *Versicolour* and *Virginica*. Four distinct features ($D = 4$) are utilised to describe every instance in the dataset (Fisher, 1936).

Characterisation

- The dataset is completely balanced ($N_1 = N_2 = N_3 = 50$). While *Setosa* class is linearly separable from the others, *Versicolour* and *Virginica* classes are not linearly separable from each other (*i.e.* overlapped classes).
- The objects in this dataset are characterised by the 4 features next listed, along with their range of values:
  - Sepal length in cm ($[4.3, 7.9]$)
  - Sepal width in cm ($[2, 4.4]$)
  - Petal length in cm ($[1, 6.9]$)
  - Petal width in cm ($[0.1, 2.5]$)
- The Euclidean distance is used as proximity measure.

[3] http://archive.ics.uci.edu/ml/datasets/Iris

Cluster analysis

As shown in Figure 5.29, three different clusters are identified by the LSS-GCSS algorithm in the *Iris* dataset:

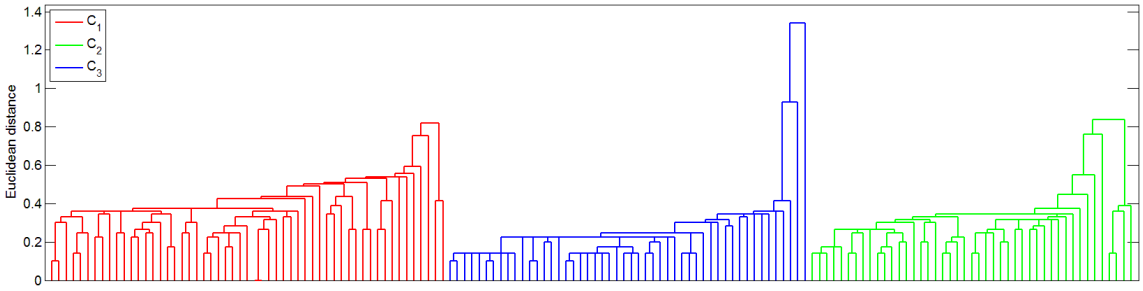


Figure 5.29: *Iris* dataset: clustering solution by LSS-GCSS.

Results

The evaluation of the obtained clustering results (see Tables 5.4 and 5.5) indicates that LSS-GCSS successfully identifies the three different types of iris plants present in the *Iris* dataset:

CI	$\overline{S}$	$CPCC$		
		$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$
0.953	0.484	0.622	0.611	0.472

Table 5.4: Validity measures for the clustering solution shown in Figure 5.29.

	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$
Setosa	0	0	50
Versicolour	6	44	0
Virginica	49	1	0

Table 5.5: Matching matrix of the clustering solution shown in Figure 5.29.

- The obtained clustering solution matches the ground truth in a high degree ($CI = 0.953$), which points out a proper identification of the real clusters present in the dataset.
- The matching matrix resulting from the calculation of CI illustrates the mapping between the three types of iris plants present in the dataset and the obtained clusters, as well as the misassignments performed by the clustering solution (7 misassigned objects out of 150). The linearly separable *Setosa* class (cluster $\mathbf{C}_3$) is perfectly identified, whereas the partial overlapping between *Versicolour* and *Virginica* classes causes the misassignments located in clusters $\mathbf{C}_2$ and $\mathbf{C}_1$, respectively.
- Again, the value of the Silhouette Coefficient ($\overline{S} = 0.484$) indicates the presence of relatively low-compacted and/or low-isolated clusters; specially $\mathbf{C}_1$ and $\mathbf{C}_2$, since $\mathbf{C}_3$ identifies a fully isolated class, which may explain the slight increase in the value of $\overline{S}$ in comparison with the previous section.
- The $CPCC$ values are lower than those obtained in the previous section, but high enough to confirm the suitability of the representation provided by the obtained dendrograms, whose shapes, again, indicate the presence of clusters of variable density.

Finally, the results of the Kruskal-Wallis test shown in Table 5.6 confirm that all clusters identify different statistical distributions of data, except for clusters $\mathbf{C}_1$ and $\mathbf{C}_2$ when considering the sepal width ($p = 0.048$), which concurs with the absence of linear separation (*i.e.* partial overlapping) between *Versicolour* and *Virginica* classes.

	$\mathbf{C}_1 - \mathbf{C}_2$	$\mathbf{C}_1 - \mathbf{C}_3$	$\mathbf{C}_2 - \mathbf{C}_3$
Sepal length in cm	$3.99\,10^{-7}$	$1.50\,10^{-18}$	$5.04\,10^{-12}$
Sepal width in cm	0.048	$1.13\,10^{-9}$	$1.03\,10^{-12}$
Petal length in cm	$1.05\,10^{-15}$	$9.12\,10^{-19}$	$3.97\,10^{-17}$
Petal width in cm	$7.88\,10^{-15}$	$4.22\,10^{-19}$	$1.51\,10^{-17}$

Table 5.6: Kruskal-Wallis' p-values in the clustering solution shown in Figure 5.29.

5.3.3 The Wisconsin Diagnostic Breast Cancer dataset

The Wisconsin Diagnostic Breast Cancer (WDBC) dataset[4] comes from data resulting from real cases of breast cancer medical diagnosis at the University of Wisconsin Hospital, which involve measurements taken according the Fine Needle Aspirate (FNA) test. This test involves fluid extraction from a breast mass using a small-gauge needle and visual inspection of the cell nuclei present in the fluid under a microscope; therefore, the data consist of diverse features of cell nuclei (size, shape, thickness, texture, area, concavity, etc.) measured from microscopic images. According to these features, the diagnosis is performed and breast masses are categorised into either benign –*i.e.* no further actions are required– or malignant –*i.e.* a breast cancer case has been diagnosed, so that the malignant mass must be excised– (Anagnostopoulos et al., 2006).

There exist two different versions of this dataset, both highly used in bioinformatics and machine learning fields: the *WDBC#1* dataset, which dates from year 1992 (see section 5.3.3.1), and the *WDBC#2* dataset, which dates from year 1995 (see section 5.3.3.2).

5.3.3.1 The *WDBC#1* dataset

Input data

The *WDBC#1* dataset[5] comprises data from 699 images belonging to two different breast mass diagnostic cases ($K_{opt} = 2$): *Benign* and *Malignant*. Every image is characterised by means of nine distinct cell nuclei features ($D = 9$) (Bennett and Mangasarian, 1992).

[4] http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
[5] http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.names

Characterisation

- It is worth noting that there are 16 objects in the dataset which present a single missing (*i.e.* unavailable) feature value. In addition, 234 objects are duplicated (*i.e.* they are identical to some other object in the dataset). Therefore, after omit both faulty and duplicated objects from the initial 699, the cluster analysis is run on the four hundred and forty-nine remaining objects ($N = 449$).

- The real clusters in the dataset (formed by objects belonging to *Benign* and *Malignant* classes) are not linearly separable and slightly unbalanced ($N_1 = 213$ and $N_2 = 236$, respectively).

- The objects in this dataset are characterised by the 9 features next listed, where the range of values is the same for all features ($\{1, 10\}$):

  - Clump thickness
  - Uniformity of cell size
  - Uniformity of cell shape
  - Marginal adhesion
  - Single epithelial cell size
  - Bare nuclei
  - Bland chromatin
  - Normal nucleoli
  - Mitoses

- All features are normalised to zero mean and unit variance prior to cluster analysis.

- The Cosine distance is used as proximity measure.

Cluster analysis

As shown in Figure 5.30, two different clusters are identified by the LSS-GCSS algorithm in the *WDBC#1* dataset:

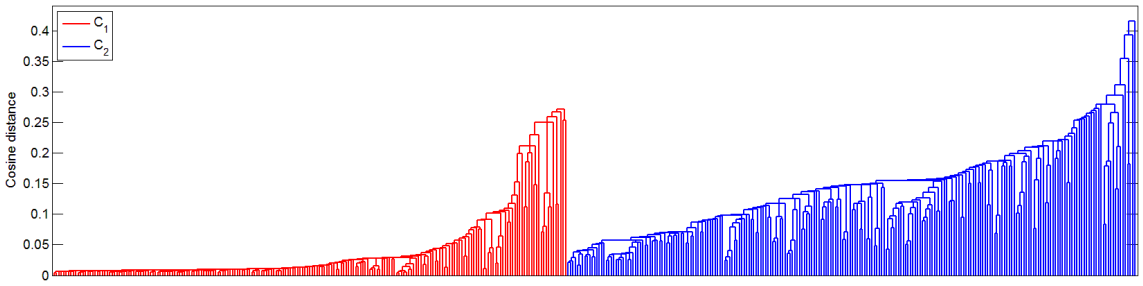


Figure 5.30: *WDBC#1* dataset: clustering solution by LSS-GCSS.

Results

The evaluation of the obtained clustering results (see Tables 5.7 and 5.8) indicates that LSS-GCSS successfully identifies the two different classes of images present in the *WDBC#1* dataset:

CI	$\overline{S}$	$CPCC$	
		$\mathbf{C}_1$	$\mathbf{C}_2$
0.947	0.661	0.776	0.508

Table 5.7: Validity measures for the clustering solution shown in Figure 5.30.

	$\mathbf{C}_1$	$\mathbf{C}_2$
Benign	201	12
Malignant	12	224

Table 5.8: Matching matrix of the clustering solution shown in Figure 5.30.

- The obtained clustering solution matches the ground truth in a high degree ($CI = 0.947$), which points out a proper identification of the real clusters present in the dataset.
- The matching matrix resulting from the calculation of CI illustrates the mapping between the two types of images present in the dataset and the obtained clusters, as well as the misassignments performed by the clustering solution (24 misassigned objects out of 449). $\mathbf{C}_1$ identifies *Benign* cases, while images resulting from *Malignant* cases tend to be grouped within $\mathbf{C}_2$. Regarding the missassignments, the 12 malign cases included in $\mathbf{C}_1$ are more dangerous than the others, since they are false negative cases from a diagnostic perspective (*i.e.* they would be malignant breast masses diagnosed as non-carcinogenic cases).
- The value of the Silhouette Coefficient ($\overline{S} = 0.661$) indicates the presence of more compacted and isolated clusters in comparison with *Wine* and *Iris* datasets.
- Again, $CPCC$ values confirm the suitability of the representation provided by the obtained dendrograms, which indicate the presence of clusters of variable density (specially $\mathbf{C}_1$).

Finally, the results of the Kruskal-Wallis test confirm that both clusters identify different statistical distributions of data, since there are significant differences ($p < 0.01$) between both clusters along all features in the dataset.

5.3.3.2 The *WDBC#2* dataset

Input data

The *WDBC#2* dataset[6] comprises data from five hundred and sixty-eight images ($N = 568$) belonging to two different breast mass diagnostic cases ($K_{opt} = 2$): *Benign* and *Malignant*. Every image is characterised by means of thirty distinct cell nuclei features ($D = 30$) (Mangasarian et al., 1995).

Characterisation

- The real clusters in the dataset (formed by objects belonging to *Benign* and *Malignant* classes) are not linearly separable and more unbalanced than in the first version of the dataset ($N_1 = 357$ and $N_2 = 212$, respectively).

[6] http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.names

- Every object in this dataset is characterised by 30 features, which result from the following process. Firstly, 10 real-valued attributes are measured for each cell nucleus present in the image. And secondly, the mean (μ), the standard deviation (σ) and the mean of the three largest values $(\mu^{(3)})$ of every attribute are calculated, giving rise to the 30 definitive features. Both the original attributes and the range of values of every final feature are next listed:
  - Radius $(\mu \in [6.98, 28.11]\,,\ \sigma \in [0.112, 2.873]\,,\ \mu^{(3)} \in [7.93, 36.04])$
  - Texture $(\mu \in [9.71, 39.28]\,,\ \sigma \in [0.36, 4.885]\,,\ \mu^{(3)} \in [12.02, 49.54])$
  - Perimeter $(\mu \in [43.79, 188.5]\,,\ \sigma \in [0.757, 21.98]\,,\ \mu^{(3)} \in [50.41, 251.2])$
  - Area $(\mu \in [143.5, 2501]\,,\ \sigma \in [6.802, 542.2]\,,\ \mu^{(3)} \in [185.2, 4254])$
  - Smoothness $(\mu \in [0.053, 0.163]\,,\ \sigma \in [0.002, 0.031]\,,\ \mu^{(3)} \in [0.071, 0.223])$
  - Compactness $(\mu \in [0.019, 0.345]\,,\ \sigma \in [0.002, 0.135]\,,\ \mu^{(3)} \in [0.027, 1.058])$
  - Concavity $(\mu \in [0, 0.427]\,,\ \sigma \in [0, 0.396]\,,\ \mu^{(3)} \in [0, 1.252])$
  - Concave points $(\mu \in [0, 0.201]\,,\ \sigma \in [0, 0.053]\,,\ \mu^{(3)} \in [0, 0.291])$
  - Symmetry $(\mu \in [0.106, 0.304]\,,\ \sigma \in [0.008, 0.079]\,,\ \mu^{(3)} \in [0.157, 0.664])$
  - Fractal dimension $(\mu \in [0.05, 0.097]\,,\ \sigma \in [0.001, 0.03]\,,\ \mu^{(3)} \in [0.055, 0.208])$
- All features are normalised to zero mean and unit variance prior to cluster analysis.
- The Cosine distance is used as proximity measure.

Cluster analysis

As shown in Figure 5.31, two different clusters are identified by the LSS-GCSS algorithm in the *WDBC#2* dataset:

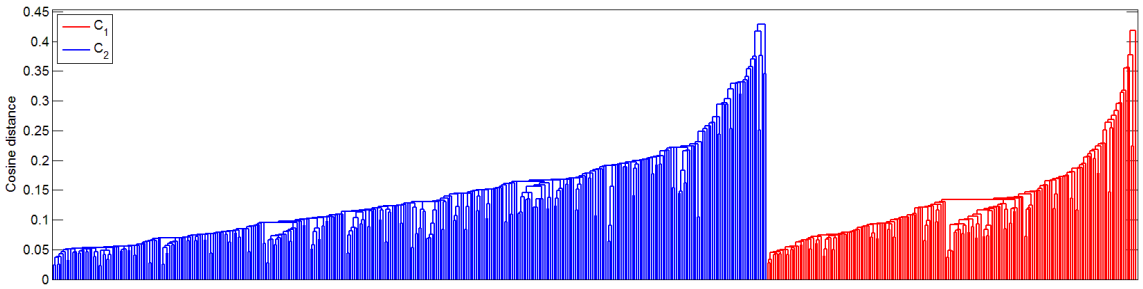


Figure 5.31: *WDBC#2* dataset: clustering solution by LSS-GCSS.

Results

The evaluation of the obtained clustering results (see Tables 5.9 and 5.10) indicates that LSS-GCSS successfully identifies the two different classes of images present in the *WDBC#2* dataset:

CI	$\overline{S}$	$CPCC$	
		$\mathbf{C}_1$	$\mathbf{C}_2$
0.94	0.456	0.42	0.459

Table 5.9: Validity measures for the clustering solution shown in Figure 5.31.

	$\mathbf{C}_1$	$\mathbf{C}_2$
Benign	349	8
Malignant	26	186

Table 5.10: Matching matrix of the clustering solution shown in Figure 5.31.

- The obtained clustering solution matches the ground truth in a high degree $(CI = 0.94)$, which points out a proper identification of the real clusters present in the dataset.
- Similarly to the previous section, the matching matrix resulting from the calculation of CI indicates that $\mathbf{C}_1$ identifies *Benign* cases, while *Malignant* cases are identified by $\mathbf{C}_2$, with 32 misassigned objects out of 568, 26 of which are false negative cases included in $\mathbf{C}_1$.
- The value of the Silhouette Coefficient $(\overline{S} = 0.456)$ indicates the presence of less compacted and isolated clusters than in the *WDBC#1* dataset.
- In addition, $CPCC$ values are lower than those obtained in the *WDBC#1* dataset, but high enough to confirm the suitability of the obtained dendrograms.

Finally, the Kruskal-Wallis statistical hypothesis test indicate the presence of significant differences between both clusters along every feature in the dataset, except for the four cases shown in Table 5.11 (*i.e.* $p < 0.01$ in 26 of the 30 features), which confirms that both clusters identify different statistical distributions of data.

Feature	p-value
Mean of the fractal dimension	0.105
Standard deviation of the texture	0.984
Standard deviation of the smoothness	0.565
Standard deviation of the symmetry	0.481

Table 5.11: Non-significant differences in the clustering solution shown in Figure 5.31.

5.3.4 The *SAD* dataset

Input Data

The *SAD* dataset[7] includes speech data from 44 male and 44 female native Arabic speakers between the ages 18 and 40, and it comprises time series of mel-frequency cepstrum coefficients (MFCC) corresponding to ten (from 0 to 9) Spoken Arabic Digits $(K_{opt} = 10)$. This dataset comes from the Laboratory of Automatic and Signals, belonging to the University of Badji-Mokhtar in Annaba, Algeria.

[7] http://archive.ics.uci.edu/ml/datasets/Spoken+Arabic+Digit

More specifically, the *SAD* dataset includes 8800 time series of MFCC taken from 88 different Arabic native speakers, each to perform 10 repetitions of 10 digits. Each time series comprise a determinate number of frames (M_i), which varies depending on the digit, the speaker and the repetition ($M_i \in \{4, 93\}$, $\forall\, i \in \{1, 8800\}$). Each frame is composed of 13 MFCC computed under the following conditions (Hammami and Bedda, 2010):

- Sampling rate: 11025 Hz
- Framing process: Hamming window
- Quantisation resolution: 16 bits
- Pre-emphasis filter: $H\,(z) = 1 - 0.97\,z^{-1}$

Therefore, the ith object in the dataset (corresponding to a given digit, a given speaker and a given repetition) is characterised by $13\,M_i$ features.

Characterisation

- Objects belonging to different users are analysed separately. Hence, 88 different datasets are generated (one per user), each including 10 repetitions of 10 digits belonging to one single user ($N = 100$) and each comprising 10 perfectly balanced clusters ($N_i = 10$, $\forall i \in \{1, 10\}$), being N_i the size of cluster $\mathbf{C}_i$.

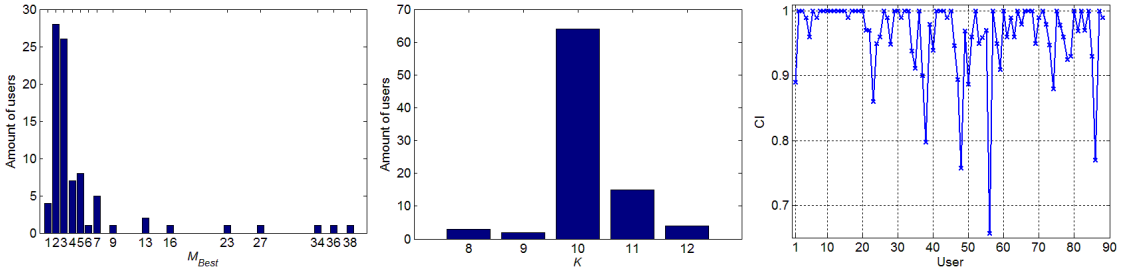
- Given a user (*i.e.* a dataset), in order for the 100 objects in that dataset to be characterised by the same number of features, a resampling process is performed over every time series (*i.e.* every object). Thus, the resampling factor applied to the ith object ($\mathbf{x}_i$) is R_i ($R_i \in \mathbb{Q}^+$), so that all objects are finally characterised by $13\,M$ features ($M = R_i\,M_i$, $\forall i \in \{1, 100\}$).

- Since the selection of the value of M can easily have a great influence on the clustering results, the study performed in the present section covers a range of values of M ($M \in \{1, 100\}$). Thus, that characterisation which leads to the best clustering results (M_{Best}) is finally selected *a posteriori* ($D = 13\,M_{Best}$). The purpose of such strategy is to separate the characterisation of this specific data from the ability of LSS-GCSS to deal with speech data, which is truly the object of the present study and which is evaluated here by minimising the negative effects of an improper data characterisation.

- The Cosine distance is used as proximity measure.

Cluster analysis

Firstly, the performance of LSS-GCSS in terms of both CI and K is shown in Figure 5.32, alongside the different values adopted by the parameter M_{Best}. Moreover, examples of clustering solutions obtained by LSS-GCSS on different users of the *SAD* dataset are shown in Figure 5.33.



(a) Histogram of M_{Best}. ***(b)*** Histogram of K. ***(c)*** CI per user.

Figure 5.32: Clustering results obtained by LSS-GCSS algorithm on the *SAD* dataset. ***(a)*** The histogram of best characterisations (M_{Best}) shows that most users are clearly located at low values (78.4% of users at $M_{Best} \in \{2,5\}$). ***(b)*** Histogram of the number of clusters (K) identified considering every user's best characterisation (M_{Best}); $K = K_{opt} = 10$ is clearly the most habitual result (72.8% of users). ***(c)*** CI values considering every user's best characterisation (M_{Best}).

Results

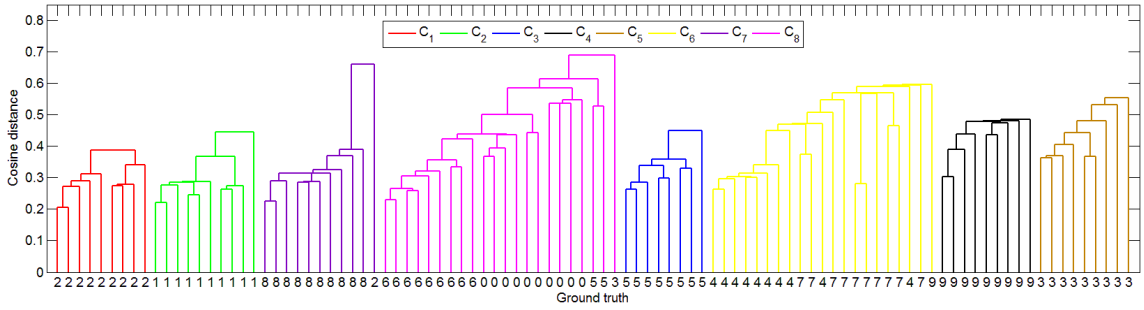
The clustering results shown in Figures 5.32b and 5.32c indicate that LSS-GCSS successfully deals with the speech data present in the *SAD* dataset. In addition, Tables 5.12 and 5.13 provide a more specific evaluation of the clustering solutions illustrated in Figure 5.33:

	User #48	User #8	User #79	User #74	**Total average**
CI	0.758	1	0.93	0.88	**0.964**
$\overline{S}$	0.334	0.77	0.619	0.658	**0.658**

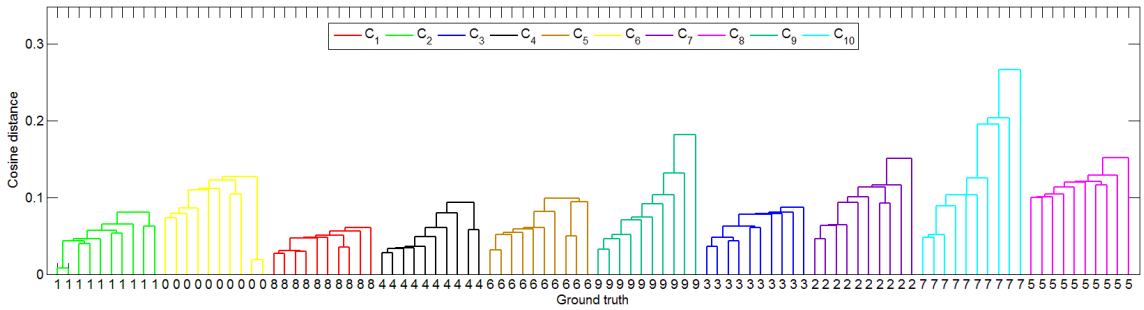
Table 5.12: Validity measures (CI and $\overline{S}$) for the clustering solutions shown in Figure 5.33. The total average values embrace all the individual results within the entire dataset (88 users).

		User #48	User #8	User #79	User #74
$CPCC$	$\mathbf{C}_1$	0.823	0.647	0.58	0.585
	$\mathbf{C}_2$	0.87	0.753	0.716	0.655
	$\mathbf{C}_3$	0.776	0.709	0.669	0.906
	$\mathbf{C}_4$	0.601	0.75	0.59	0.967
	$\mathbf{C}_5$	0.688	0.63	0.56	0.742
	$\mathbf{C}_6$	0.719	0.679	0.63	0.904
	$\mathbf{C}_7$	0.916	0.671	0.712	0.9
	$\mathbf{C}_8$	0.846	0.661	0.82	0.756
	$\mathbf{C}_9$	-	0.855	0.694	0.67
	$\mathbf{C}_{10}$	-	0.816	0.833	0.825
	$\mathbf{C}_{11}$	-	-	0.898	0.8
	$\mathbf{C}_{12}$	-	-	-	0.614

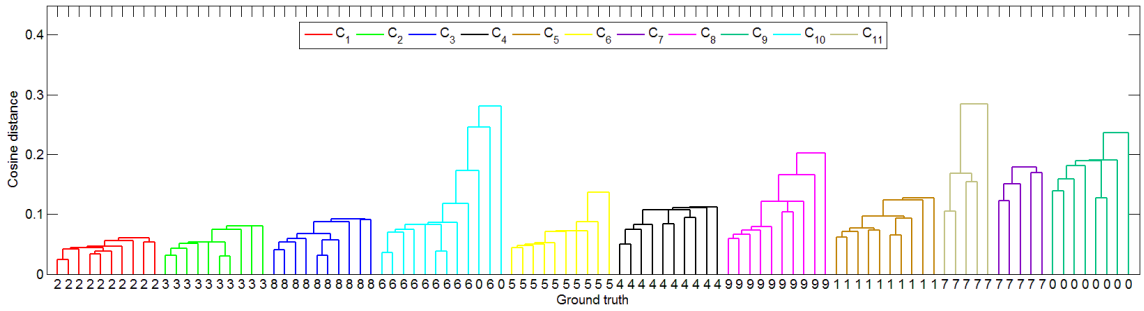
Table 5.13: Validity measures ($CPCC$) for the clustering solutions shown in Figure 5.33.



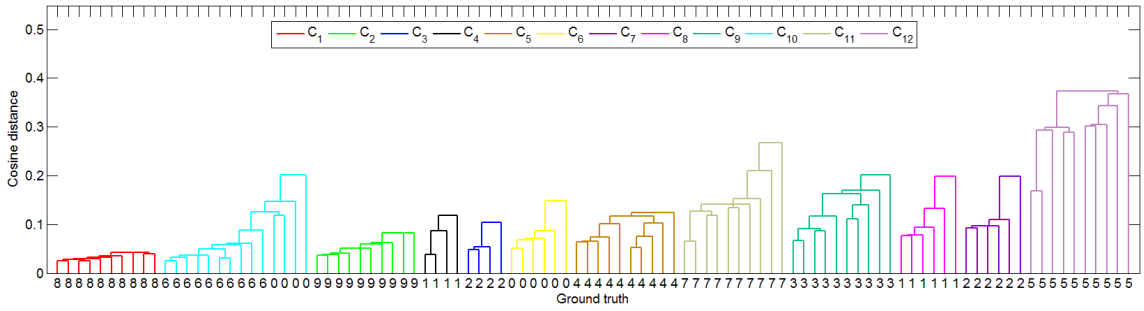
(a) User #48: LSS-GCSS dendrograms. Clusters $\mathbf{C}_6$ and $\mathbf{C}_8$ include digits 4 and 7, and 0 and 6, respectively.



(b) User #8: LSS-GCSS dendrograms. The ground truth is matched 100%.



(c) User #79: LSS-GCSS dendrograms. Digit 7 is identified by means of $\mathbf{C}_7$ and $\mathbf{C}_{11}$.



(d) User #74: LSS-GCSS dendrograms. Digits 1 and 2 are identified by means of $\mathbf{C}_4$ and $\mathbf{C}_8$, and $\mathbf{C}_3$ and $\mathbf{C}_7$, respectively.

Figure 5.33: *SAD* dataset: clustering solutions by LSS-GCSS. Ground truth's cluster labels are shown on the x-axis and directly indicate the digit (from 0 to 9) every object actually belongs to.

- The total CI averaged-value (considering all users) equals 0.964, which certainly indicates a highly successful behaviour of LSS-GCSS algorithm in global terms. In addition, Figure 5.32c shows that LSS-GCSS reaches a CI value greater that 0.9 in 90% of users.

- The real number of clusters is successfully estimated $(K = 10)$ in 72.8% of users. Furthermore, most cases with $K > 10$ are caused by some specific digits being identified by the

combination of two different clusters, which are very interesting quasi-optimal clustering results, since they avoid any confusion between different digits (see Figures 5.33c and 5.33d by way of example).

- Although the Silhouette Coefficient $(\overline{S})$ may adopt significantly different values depending on the user (see Table 5.12), its averaged-value considering all users $(\overline{S} = 0.658)$ suggests the presence of relatively high-compacted and/or high-isolated clusters in many of the datasets.
- Despite the differences in the ranges of their values $(CI \in [0.656, 1]\,,\ \overline{S} \in [0.334, 0.807])$, CI and $\overline{S}$ present some agreement. Considering the pair of variables formed by the values of CI and $\overline{S}$ along the 88 users in the dataset (under every user's best characterisation), respectively, Kendall's non-parametric statistical test (Kendall, 1938) indicates a significant degree of statistical dependency between both variables: $\tau = 0.412\ (p\text{-value} = 9.85\,10^{-8})$.
- Finally, Table 5.13 shows that $CPCC$ values tend to be high regardless of the user (probably due to the small size of all clusters, in general terms), which confirms the quality of the obtained dendrograms.

5.3.5 The *MiniNews* dataset

Input data

The *MiniNews* (or MiniNewsgroups) dataset[8] is a subset of the 20 Newsgroups document collection that comprises 100 text documents of each of the following newsgroups (or topics) –*i.e.* it comprises 2000 documents and 43063 terms (or words)– (Zha et al., 2001):

- **NG1**: *alt.atheism*
- **NG2**: *comp.graphics*
- **NG3**: *comp.os.ms-windows.misc*
- **NG4**: *comp.sys.ibm.pc.hardware*
- **NG5**: *comp.sys.mac.hardware*
- **NG6**: *comp.windows.x*
- **NG7**: *misc.forsale*
- **NG8**: *rec.autos*
- **NG9**: *rec.motorcycles*
- **NG10**: *rec.sport.baseball*
- **NG11**: *rec.sport.hockey*
- **NG12**: *sci.crypt*
- **NG13**: *sci.electronics*
- **NG14**: *sci.med*
- **NG15**: *sci.space*
- **NG16**: *soc.religion.christian*
- **NG17**: *talk.politics.guns*
- **NG18**: *talk.politics.mideast*
- **NG19**: *talk.politics.misc*
- **NG20**: *talk.religion.misc*

[8] http://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups

The most typical representation of the textual information is based on considering documents as term vectors (Salton and Buckley, 1988); *i.e.* a document is just a collection of terms (or bag-of-words), where each term is weighted as follows (Joachims, 1996):

$$x_{i_j} = tf_{ij} \;\; idf_j = tf_{ij} \;\; \log\left(\frac{D}{n_j}\right), \forall\, i \in \{1, N\}, \forall\, j \in \{1, D\} \tag{5.4}$$

where tf_{ij} is the number times the jth term occurs in the ith document ($\mathbf{x}_i$) and n_j is the number of documents in which the jth term occurs at least once (*i.e.* the document frequency of the term). This representation gives rise to a document-by-term matrix $\mathbf{X}$ (*i.e.* a dataset), where and N and D are the number of documents (*i.e.* objects) and terms (*i.e.* features), respectively, in the dataset.

Characterisation

- The present study analyses the performance of LSS-GCSS in the face of all the binary clustering problems of variable difficulty present in the *MiniNews* dataset (Srinivasan, 2002); *i.e.* different newsgroups present different degrees of semantic overlapping (*e.g.* NG1 and NG2 are well-separated topics, whereas NG18 and NG19 are highly overlapped). Hence, 190 different datasets are analysed, resulting from the 190 possible couples of newsgroups resulting from the *MiniNews* dataset: couple #1 (NG1 & NG2), couple #2 (NG1 & NG3)... couple #190 (NG19 & NG20). Thus, each single dataset comprises two hundred documents grouped into two perfectly balanced clusters ($N = 200,\ K_{opt} = 2,\ N_1 = N_2 = 100$).

- Typically, the document-by-term matrix is preprocessed in order to obtain more suitable representation spaces for the textual information, so that the characteristics relative to the topics the documents belong to are more properly represented. Thus, a D-dimensional space is built from the original document-by-term matrix in order to make easier to identify documents according to their content (*i.e.* to distinguish among documents belonging to different topics). In this context, two main strategies are usually followed to perform this dimensionality reduction (Sebastiani, 2002):

  - **Term selection**. A subset of terms from the original collection is selected in order to represent documents in a more proper space of terms. Among many others, the most typical term selection technique consists of selecting those terms whose document frequency is enclosed between document frequency thresholds β_l and β_u, respectively ($\beta_l \leq n_j \leq \beta_u$).

  - **Term extraction**. Mainly due to the pervasive problems of linguistic situations like polysemy, homonymy and synonymy, the terms by themselves may not be optimal dimensions for document content representation. Hence, the term extraction strategy involves engendering a new representation space by means of a completely new set of synthetic terms, which are generated from the original terms and which more properly represent the underlying latent semantic structure of the data.

Among others, the most typical term extraction technique is known as Latent Semantic Indexing (LSI) (Deerwester et al., 1990), which generates the new representation space by performing a Principal Component Analysis (PCA) implemented through the Single Value Decomposition (SVD) of the original document-by-term matrix. Every dimension of the new semantic space is identified with the principal components resulting from LSI, so that new dimensions result from linear combinations of terms weighted through the SVD. Thus, the dimensionality of the new space corresponds to the number of principal components ($D = \alpha$), which is required to be selected and which is usually much lower than the original number of terms.

- Previous works present in the literature indicate that the accuracy of clustering solutions in this context depends in extremely high degree on a proper characterisation of the textual information (Cobo et al., 2006; Sevillano et al., 2006a). In addition, optimal text representations are difficult to be determined beforehand and may also vary from one clustering problem to another (Sevillano et al., 2006b). This particular issue is analysed more deeply in the literature, where strategies for robust document clustering based on a consensus clustering approach have been proposed (Sevillano et al., 2007a,b).

  Nonetheless, the object of the present study is not the optimal representation of textual information, but the ability of LSS-GCSS to deal with text data. Therefore, in order to minimise the negative effects of an improper data characterisation, the study performed in the present section covers a range of values of parameters β_l, β_u and α ($\beta_l \in \{1, 10\}$, $\beta_l \in \{10, 200\}$ and $\alpha \in \{1, 30\}$). Thus, similarly to the previous section, that characterisation –either by term selection (β_l, β_u), or by term extraction (α)– which leads to the best clustering results is finally selected *a posteriori*.

- The Cosine distance is used as proximity measure.

Cluster analysis

The performance of LSS-GCSS in terms of both CI throughout the 190 possible couples of newsgroups in the *MiniNews* dataset is shown in Figure 5.34:

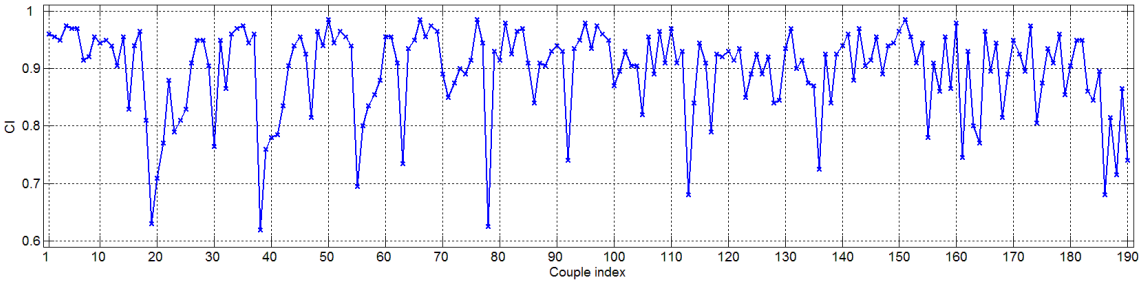
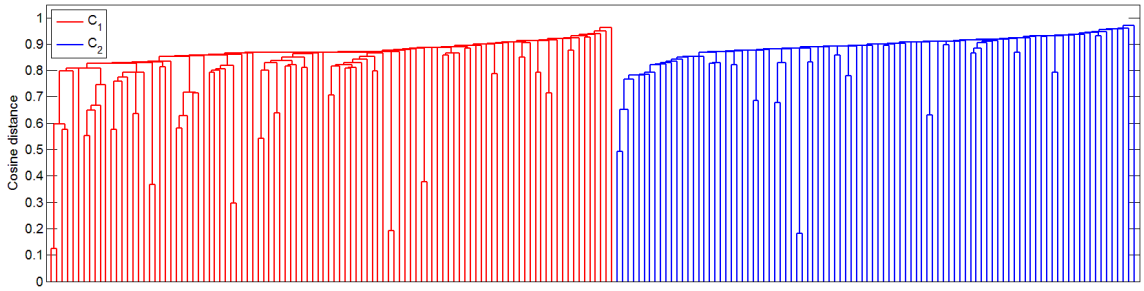
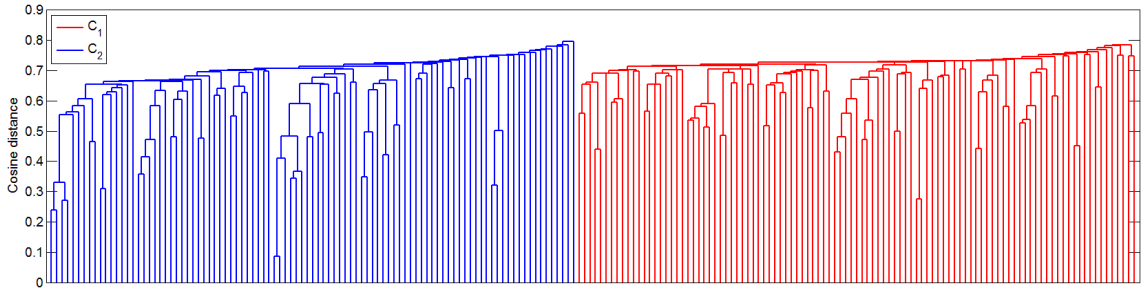


Figure 5.34: Clustering results obtained by LSS-GCSS on the *MiniNews* dataset. CI values per couple of newsgroups, considering the best characterisation (*i.e.* the best clustering result) for every couple.

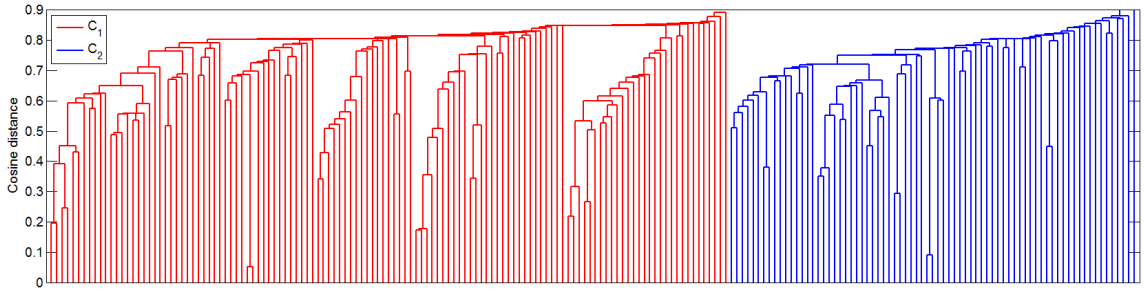
Moreover, examples of clustering solutions obtained by LSS-GCSS in the face of different couples of newsgropus –each under its best characterisation– of the *MiniNews* dataset are shown in Figure 5.35. Specifically, three binary clustering problems of incremental difficulty are illustrated (Srinivasan, 2002): NG1 and NG2 (*alt.atheism* and *comp.graphics*) are well-separated topics (see Figure 5.35a), NG10 and NG11 (*rec.sport.baseball* and *rec.sport.hockey*) are relatively overlapped topics (see Figure 5.35b), and, finally, NG18 and NG19 (*talk.politics.mideast* and *talk.politics.misc*) are highly overlapped topics (see Figure 5.35c).



(a) Couple #1 (NG1 & NG2): LSS-GCSS dendrograms ($\beta_l = 1,\ \beta_u = 40,\ D = 10095$).



(b) Couple #136 (NG10 & NG11): LSS-GCSS dendrograms ($\beta_l = 10,\ \beta_u = 20,\ D = 321$).



(c) Couple #188 (NG18 & NG19): LSS-GCSS dendrograms ($\beta_l = 5,\ \beta_u = 20,\ D = 1273$).

Figure 5.35: *MiniNews* dataset: clustering solutions by LSS-GCSS. The specifics of each couple's best characterisation are detailed in the captions.

Results

The clustering results shown in Figure 5.34 indicate that LSS-GCSS presents a successful performance when handling binary clustering problems of text data. In addition, Tables 5.14 and 5.15 provide a more specific evaluation of the clustering solutions illustrated in Figure 5.35.

In addition, more specific conclusions can be drawn:

	Couple #1	Couple #136	Couple #188	**Total average**
CI	0.96	0.725	0.715	**0.896**
$\overline{S}$	0.013	0.017	0.016	**0.0161**

Table 5.14: Validity measures (CI and $\overline{S}$) for the clustering solutions shown in Figure 5.35. The total average values embrace all the individual results within the entire dataset (190 couples of newgroups).

		Couple #1	Couple #136	Couple #188
$CPCC$	$\mathbf{C}_1$	0.621	0.367	0.724
	$\mathbf{C}_2$	0.578	0.502	0.492

Table 5.15: Validity measures ($CPCC$) for the clustering solutions shown in Figure 5.35.

- The total CI averaged-value (considering all couples) equals 0.896, which certainly indicates a successful behaviour of LSS-GCSS algorithm in global terms. More specifically, the best clustering result corresponds to couple #50 (NG3 and NG16, $\beta_l = 1$, $\beta_u = 100$, $D = 11051$, $K = 2$, $CI = 0.985$), whereas the worst clustering result corresponds to couple #78 (NG5 and NG13, $D = \alpha = 5$, $K = 2$, $CI = 0.6250$).

- The real number of clusters is successfully estimated ($K = 2$) by LSS-GCSS in 189 out of the 190 possible couples (the exception is couple #155: NG12 and NG13, $D = \alpha = 20$, $K = 3$, $CI = 0.78$), which is a highly successful result. More specifically, LSS-GCSS tends to behave better with term selection ($K = 2$ in 184 couples) than with term extraction ($K = 2$ in 124 couples).

- According to the preprocessing stage, term selection leads to the best characterisation in 141 couples (if only term selection had been considered, the total averaged CI would have equalled 0.882), whereas term extraction leads to the best characterisation in the 49 remaining couples (if only term extraction had been considered, the total averaged CI would have equalled 0.836).

- The Silhouette Coefficient $(\overline{S})$ adopts significantly low values regardless of the couple (see Table 5.14), which clearly indicates that text data tend to be sparse and to group into certainly low-compacted and low-isolated clusters.

- Table 5.13 shows that, although they may vary depending on the couple of newsgroups, $CPCC$ values are high enough to confirm the suitability of the representation provided by the obtained dendrograms, whose ranges of proximity values also indicate that objects are sparsely arranged within their clusters (see Figure 5.35).

- It is worth noting that the performance of LSS-GCSS in the face of binary clustering problems of text data is essentially the same regardless of which distance function (either Cosine or Euclidean, which has been also tested) is used as proximity measure.

- The performance of LSS-GCSS has been also tested in the face of combinations of more than two newsgroups ($K_{opt} > 2$) and it suffers from a significant decrease, which is in agreement with the behaviour of LSS-GCSS in the presence of multiple low-compacted and low-isolated clusters (see section 5.2.4).

- Finally, the obtained clustering results in the present section indicate, in agreement with the literature, that the accuracy of clustering solutions highly depends on the characterisation of the data. This fact suggests that, regarding the characterisation of textual data, it might be interesting to consider strategies beyond the typical bag-of-words approach (*e.g.* features related to other grammatical units –groups of words, sentences, subordinate sentences– or grammatical marks –periods, commas, colons, semicolons–) in order to give rise to more compacted and isolated clusters, which might be helpful in the subsequent cluster analysis, regardless of what specific clustering algorithm is used.

5.4 Comparative study between LSS-GCSS and other clustering algorithms

The main goal of the present section is to conduct a comparison between LSS-GCSS and other clustering algorithms in terms of performance. To that effect, a variety of clustering algorithms (both partitional and hierarchical, both parameter-dependent and parameter-free) is tested by a set of both synthetic and real datasets selected from the two previous sections.

Thus, on the one hand, the present comparative study comprises the following clustering scenarios:

- ***2unif***: 100 instances of the *2unif* dataset (see section 5.2.2.1), each comprising two touching uniformly-distributed clusters ($K_{opt} = 2$, $N_1 = N_2 = 500$, $D = 2$, $d = 0.05$).

- ***2Gauss***: 100 instances of the *2Gauss* dataset (see section 5.2.2.2), each comprising two partially overlapped Gaussian clusters ($K_{opt} = 2$, $N_1 = N_2 = 500$, $D = 2$, $n = 4$).

- ***2bars***: 100 instances of the *2bars* dataset (see section 5.2.2.3), each comprising two touching bar-shaped clusters ($K_{opt} = 2$, $N_1 \in \{642, 706\}$, $N_2 \in \{651, 705\}$, $D = 2$, $\Delta_x = -0.2$).

- ***2ubGauss***: 100 instances of the *2ubGauss* dataset (see section 5.2.3), each comprising two slightly-overlapped unbalanced Gaussian clusters ($K_{opt} = 2$, $R = 5$, $N_1 = 100$, $N_2 = 500$, $D = 2$, $n = 6$).

- ***Munif***: 100 instances of the *Munif* dataset (see section 5.2.4), each comprising twenty well-separated uniformly-distributed clusters ($M = 20$, $K_{opt} = 20$, $N_i = 100$, $D = 2$, $d = 0.25$).

- ***3rings***: 100 instances of the *3rings* dataset (see section 5.2.5.1), each comprising three touching ring-shaped clusters ($K_{opt} = 3$, $N_1 = N_2 = N_3 = 500$, $D = 2$, $d = 0.15$).

- ***2spirals***: 100 instances of the *2spirals* dataset (see section 5.2.5.2), each comprising two touching spiral-shaped clusters ($K_{opt} = 2$, $N_1 = 600$, $N_2 = 550$, $D = 2$, $\alpha = \frac{\pi}{2}$).

- ***Mix#1***, ***Mix#2*** and ***Mix#3***: Three different synthetic datasets that comprise a mix of arbitrary-shaped clusters (see section 5.2.6).

- ***Wine***, ***Iris***, ***WDBC#1*** and ***WDBC#2***: Four benchmark datasets corresponding to real-world clustering problems (see sections 5.3.1, 5.3.2, 5.3.3.1 and 5.3.3.2, respectively, for further details).

- ***SAD***: 88 instances of the *SAD* dataset, each corresponding to a different speaker (see section 5.3.4 for further details).

- ***MiniNews***: 190 instances of the *MiniNews* dataset, each corresponding to a different couple of newsgroups (see section 5.3.5 for further details).

On the other hand, the present comparative study covers a diversity of clustering algorithms. It comprises, among others, the clustering algorithms most widely used in practice, which includes the majority of approaches to clustering that deal with the estimation of the number of clusters –*i.e.* relative validity approaches, self-refining consensus approaches and model-based approaches–, regardless of whether they are parameter-free or not (see section 2.5.1 for further details):

- **k-means**: Centre-based HPC algorithm that requires both the number of clusters (K) and the initialisation of its K centroids (Forgy, 1965; MacQueen, 1967). Hence, in order to optimise its performance, it is provided with the real number of clusters ($K = K_{opt}$) and it is run ten times with ten different random initialisations (the best clustering solution is selected *a posteriori*) on each instance of every dataset.

- **EM**: Model-based HPC algorithm that requires an initial clustering solution, typically provided by the k-means algorithm (see section 2.5.1.5 for further details). To avoid local minima, the k-means algorithm is run ten times with ten different random initialisations and the highest likelihood solution is used in order to begin the EM estimation.

  In addition, EM clustering algorithm may or may not require the value of K, since it can be initialised either once for a specific value of K, or several times for a range of values of K and select the optimal initialisation (and therefore the final value of K) according to a Monte Carlo cross-validation method (Smyth, 1996). In the present study, EM is provided with the real number of clusters ($K = K_{opt}$).

- **x-means**: Model-based HPC algorithm, which, after being initialised by multiple k-means clustering solutions with different values of K, selects the best one of them according to the BIC function 2.5.1.5. It requires both the K_{min} and K_{max} values that enclose the range of possible values of K ($K \in \{K_{min}, K_{max}\}$). In the present study, both parameters are set so that the resulting range of values of K includes the real number of clusters in all the tested scenarios ($K_{min} = 1$, $K_{max} = 25$).

- **SL**, **CL**, **UPGMA**, **WPGMA**, **UPGMC**, **WPGMC**, and **Ward's**: Both graph-based (SL, CL, UPGMA and WPGMA) and geometric (UPGMC, WPGMC and Ward's) basic AHC algorithms that do not require any initial parametrisation (see sections 3.1.1 and 3.1.2, respectively, for further details). Nonetheless, these clustering algorithms result in a dendrogram, which needs to be postprocessed to give rise to a HPC solution of K clusters. In the present study, final HPC solutions are obtained from dendrograms by means of the ZIC approach, which requires two parameters –depth (δ) and inconsistency threshold (ι_{th})– to be defined (see section 3.1.3 for further details). Thus, in order to optimise the performance of these algorithms, a range of values of every parameter is covered ($\delta \in \{1, 50\}$; $\iota_{th} \in \{\iota_1, \iota_{20}\}$, where $\iota_1 ... \iota_{20}$ are the inconsistency values of the 20 most inconsistent links in the dendrogram) and the best clustering solution is selected *a posteriori*.

- **SL-DID** and **LSS-GCSS**: Parameter-free AHC algorithms that do not require any user intervention (see sections 3.2.1 and 4.2 for further details).

It is worth noting that the parameter-dependent clustering algorithms included in the present study certainly enjoy a comparative advantage, since the negative effects of an improper parametrisation are eliminated by selecting *a posteriori* the best possible clustering solution each of them can provide in every scenario. This strategy allows to evaluate them at their maximum performance, which in its turn includes the best possible performance of any approach to the estimation of K_{opt} based on the analysis of a variety of clustering solutions provided by any of these algorithms (see section 2.5.1 for further details).

Moreover, in order for every tested clustering algorithm to perform at its best in all the scenarios, the same data characterisation procedures carried out in sections 5.2 and 5.3 on every specific dataset are applied in the present section to every dataset and every clustering algorithm. Additionally, the same distance functions used in sections 5.2 and 5.3 as proximity measures are also kept in the present section, except for the x-means algorithm, which has reported its best clustering results by means of the Euclidean distance in all the clustering scenarios.

Hence, the results of the present comparative study are shown in Table 5.16. Regarding the validation procedure of the clustering results, every clustering solution obtained in the present section is compared with the ground truth solution and evaluated by means of the Consistency Index (CI) (see section 2.4.1 for further details).

	$\mathbf{K_{opt}}$	**k-means**	**EM**	**x-means**	**SL**	**CL**	**UPGMA**	**WPGMA**	**UPGMC**	**WPGMC**	**Ward's**	**SL-DID**	**LSS-GCSS**
2unif	2	**1** (**2**)	0.941 (2)	0.5 (1)	0.915 (2)	0.946 (2)	0.925 (2)	0.913 (2)	0.956 (2)	0.898 (2)	0.939 (2)	0.501 (1)	0.999 (2)
2Gauss	2	0.976 (2)	0.976 (2)	0.5 (1)	0.753 (50)	0.934 (2)	0.961 (2)	0.867 (2)	0.965 (2)	0.877 (2)	0.954 (2)	0.501 (2)	**0.996** (**2**)
2bars	2	0.792 (2)	0.602 (2)	0.51 (1)	0.879 (38)	0.582 (3)	0.584 (3)	0.577 (3)	0.562 (3)	0.612 (2)	0.597 (2)	0.51 (1)	**0.98** (**2**)
2ubGauss	2	**0.999** (**2**)	**0.999** (**2**)	**0.999** (**2**)	0.994 (2)	0.979 (2)	0.998 (2)	0.995 (2)	**0.999** (**2**)	0.946 (2)	**0.999** (**2**)	0.949 (3)	0.912 (2)
Munif	20	0.921 (20)	0.389 (20)	0.05 (1)	**1** (**20**)	0.986 (20)	0.986 (20)	0.966 (20)	0.997 (20)	0.919 (20)	0.989 (20)	0.999 (21)	**1** (**20**)
3rings	3	0.449 (3)	0.477 (3)	0.333 (1)	0.875 (8)	0.549 (3)	0.525 (5)	0.507 (5)	0.546 (5)	0.49 (5)	0.576 (4)	0.368 (1)	**0.926** (**3**)
2spirals	2	0.499 (2)	0.514 (2)	0.522 (1)	0.548 (2)	0.409 (5)	0.5 (2)	0.516 (2)	0.463 (3)	0.481 (3)	0.505 (2)	0.522 (1)	**0.992** (**2**)
Mix#1	7	0.637 (7)	0.662 (7)	0.253 (1)	0.996 (8)	0.608 (12)	0.588 (7)	0.664 (6)	0.672 (7)	0.533 (7)	0.662 (5)	0.638 (3)	**1** (**7**)
Mix#2	6	0.507 (6)	0.593 (6)	0.365 (1)	**1** (**6**)	0.514 (7)	0.565 (9)	0.485 (7)	0.531 (6)	0.468 (14)	0.524 (10)	0.666 (3)	**1** (**6**)
Mix#3	5	0.997 (5)	0.86 (5)	0.313 (1)	**1** (**5**)	0.942 (4)	0.948 (4)	0.886 (17)	0.948 (4)	0.948 (4)	**1** (**5**)	0.998 (6)	**1** (**5**)
Wine	3	0.966 (3)	**0.972** (**3**)	0.949 (3)	0.691 (45)	0.837 (3)	0.916 (8)	0.91 (7)	0.904 (8)	0.77 (9)	0.933 (3)	0.399 (1)	0.938 (3)
Iris	3	0.893 (3)	0.907 (3)	0.667 (2)	0.827 (18)	0.84 (3)	0.88 (4)	0.9 (3)	0.88 (4)	0.767 (8)	0.893 (3)	0.667 (2)	**0.953** (**3**)
WDBC#1	2	0.927 (2)	0.951 (2)	0.929 (2)	0.915 (21)	0.92 (3)	0.942 (2)	0.9 (4)	0.944 (2)	0.862 (8)	0.953 (2)	**0.955** (**2**)	0.947 (2)
WDBC#2	2	0.917 (2)	0.912 (2)	0.344 (8)	0.752 (105)	0.752 (4)	**0.942** (**2**)	0.745 (2)	0.875 (2)	0.689 (11)	0.935 (2)	0.627 (1)	0.94 (2)
SAD	10	**0.989** (**10**)	0.917 (10)	0.802(10)	0.968 (10)	0.974 (10)	0.976 (10)	0.979 (10)	0.972 (10)	0.972 (10)	0.985 (10)	0.81 (9)	0.964 (10)
MiniNews	2	**0.971** (**2**)	0.512 (2)	0.5 (1)	0.806 (3)	0.804 (2)	0.9 (2)	0.901 (2)	0.86 (2)	0.826 (3)	0.922 (2)	0.622 (2)	0.896 (2)

Table 5.16: A comparison among different clustering algorithms in terms of performance. The $\mathbf{K_{opt}}$ column indicates the real number of clusters in each clustering scenario. CI values are averaged along the 100 instances of every dataset, except for ***Mix#1***, ***Mix#2***, ***Mix#3***, ***Wine***, ***Iris***, ***WDBC#1*** and ***WDBC#2*** scenarios. Similarly, the values in parenthesis correspond to the number of clusters most frequently found by each algorithm (the mode of K along the 100 instances of every dataset), except for **k-means** and **EM**, which are always fed with $K = K_{opt}$. The values emphasised in bold characters refer to the best clustering results obtained in each clustering scenario.

A detailed analysis of the obtained results allows to draw the following conclusions:

- Firstly, LSS-GCSS certainly proves to be the best overall method considering the whole set of clustering scenarios tested: it always found the real number of clusters, it always performs beyond a $0.9\ CI$ value (except in the *MiniNews* scenario, with a $0.896\ CI$) and it reaches the best clustering results in absolute terms on nine of the sixteen scenarios tested.

- Regarding the estimation of the number of clusters, LSS-GCSS clearly outperforms SLDID and x-means algorithms, which are able to properly found the real number of clusters on three and four of the scenarios, respectively. Its performance on this particular issue is even better than that of the basic AHC algorithms, whose best possible clustering solutions selected *a posteriori* do not include the correct number of clusters on many of the scenarios.

- Additionally, LSS-GCSS also outperforms k-means and EM algorithms in spite of working under worse conditions, since both HPC methods are always fed with the real number of clusters and released from the possible negative effects of a bad initialisation (specially in the case of k-means). Specifically, LSS-GCSS beats k-means and EM algorithms on eleven and thirteen of the scenarios, respectively, whereas k-means and EM are able to reach the best clustering results in absolute terms on four and two of the scenarios, respectively.

- In the particular comparison between LSS-GCSS and SLDID, the former always outperforms the latter, except on the *WDBC#1* scenario, where SLDID reaches the best clustering results in absolute terms and slightly beats LSS-GCSS. Regarding scenarios with touching and overlapping clusters (*e.g. 2Gauss*, *2bars*, *3rings* and *Iris*), the present study clearly illustrates how LSS-GCSS overcomes the most remarkable lack of SLDID.

- LSS-GCSS reaches its optimum performance on scenarios with concentric- and arbitrary-shaped clusters (*3rings*, *2spirals*, *Mix#1*, *Mix#2* and *Mix#3*), as well as in the face of some of the scenarios that present touching and overlapping clusters (*2unif*, *2Gauss*, *2bars* and *Iris*). In addition, it is also noteworthy the case of the *Munif* scenario, where LSS-GCSS and SL reach a perfect performance –since clusters are separated enough (see section 5.2.4 for further details)– and outperform the rest of methods, which proves that handling a high number of clusters can be troublesome for many clustering algorithms even working on optimal conditions (*i.e.* knowing the true number of clusters *a priori* and not depending on parametrisation issues).

  On the contrary, the hardest scenarios for LSS-GCSS are *2ubGauss*, which is both plausible and expectable, considering the highly-unbalanced overlapped clusters this scenario presents (see section 5.2.3 for further details), and *MiniNews*, since knowing in advance the real number of clusters is truly a great advantage when confronting such a difficult clustering problem (see section 5.3.5 for further details). Nonetheless, despite this latter fact, LSS-GCSS successfully estimates the real number of clusters in the *MiniNews* scenario, outperforming seven of the eleven clustering algorithms included in the study.

- The results corresponding to the basic AHC algorithms reveals that they are far from outperforming LSS-GCSS in overall terms, even enjoying the comparative advantage of considering only the best clustering results they are able to reach on every scenario, which are selected *a posteriori*. In addition, it is also shown that, in case a cluster analysis is based on a basic AHC algorithm, a proper choice of the specific AHC method (SL, CL, UPGMA, Ward's, etc.) is crucial to achieve success, since their performance may come to depend to a high extent on the nature of the clustering scenario (*e.g.* UPGMA, UPGMC and Ward's are more successful than the rest of basic AHC methods to deal with overlapped Gaussian clusters; only SL is able to properly identify arbitrary-shaped clusters; and all basic AHC methods tend to fail when handling touching bar-shaped clusters).

- Finally, it is certainly revealing to compare between the performances of k-means and x-means algorithms (the former clearly outperforms the latter, which is totally understandable), since it illustrates both the importance of knowing in advance the real number of clusters and the high influence the initialisation of a clustering algorithm has over the quality of the final clustering solution.

5.5 Discussion

The main conclusions drawn from the experimental study performed in the present chapter are next detailed:

- The LSS-GCSS algorithm is able to properly deal with a wide variety of datasets without requiring either prior knowledge about the data or any user intervention. This successful behaviour includes both appropriately finding the real number of clusters and resulting a clustering solution either highly similar or equal to the ground truth, regardless of the nature of the clusters present in the data.

  Specifically, LSS-GCSS proves to be able to identify the presence of randomly generated data, without forcing a cluster structure in the absence of it –specially in the face of Gaussian clusters and inasmuch as both size and dimensionality of data increase– (see section 5.2.1); to distinguish between touching and overlapped clusters of different distribution, density and nature (see sections 5.2.2, 5.3.2 and 5.3.3); to deal with unbalanced clusters (see sections 5.2.3, 5.2.6, 5.3.1 and 5.3.3); to identify the presence of multiple clusters in a single dataset (see sections 5.2.4, 5.2.6 and 5.3.4); to identify clusters regardless of their shape (see sections 5.2.5 and 5.2.6); and to handle data of different nature and origin (see section 5.3).

- The main limitations of the LSS-GCSS algorithm are due to either the presence of highly-unbalanced and not well-separated clusters, or the presence of a high number of not well-separated clusters.

On the one hand, the ability of LSS-GCSS to properly handle unbalanced clusters depends on their separation; *i.e.* distinguishing between highly-unbalanced clusters is a problem for LSS-GCSS, unless separation between clusters is large enough (see section 5.2.3 for further details). Specifically, separation between clusters is required to increase from unbalancing ratios equal to or greater than three ($N_2 \geq 3\,N_1$) in order for LSS-GCSS to be able to distinguish between clusters (see Figures 5.16b–5.16i). Nonetheless, LSS-GCSS easily deals with lower unbalancing ratios (see sections 5.2.6 and 5.3.1), even when clusters are partially overlapped (see Figures 5.16a and 5.17a, and section 5.3.3).

On the other hand, the ability of LSS-GCSS to properly handle a high number of clusters also depends on how well separated they are (see section 5.2.4 for further details). Specifically, separation between clusters is required to increase in order for LSS-GCSS to be able to distinguish between multiple clusters, specially from a number of clusters equal to or greater than six ($K_{opt} \geq 6$) (see Figure 5.18). Nonetheless, this limitation does not prevent LSS-GCSS from being able to identify multiple clusters in a variety of clustering scenarios, both synthetic (see Figure 5.20 and section 5.2.6) and real (see section 5.3.4). Additionally, it is worth noting that properly identifying a high number of clusters in a single dataset is not an exclusive problem of LSS-GCSS algorithm, since it can be troublesome even if the real number of clusters is known in advance (see ***Munif*** scenario in Table 5.16).

- In the context of a wide variety of clustering scenarios, the LSS-GCSS algorithm outperforms in overall terms the clustering algorithms most commonly used in practice, regarding both the ability to estimate the real number of clusters and the validity of their respective clustering solutions (see section 5.4).

Thus, as a consequence of both the conclusions reached in previous chapters of this thesis (see sections 2.6 and 3.3 for further details) and the results obtained in the experimental study carried out in the present chapter, it can be concluded that the task of obtaining a proper estimation of the true number of clusters in a dataset is more reliably performed by LSS-GCSS that by any of the previous AHC methods, specially when no knowledge about the nature of the clustering scenario is available in advance.

On the one hand, aside from the fact that they involve solving a parameter-dependent problem, basic AHC methods do not lead to successful results in determinate clustering scenarios and their behaviour may vary to a high extent from one scenario to another. On the other hand, SLDID is a well-behaved AHC algorithm that allows working under parameter-free conditions, but its performance severely decreases in the face of both touching and overlapped clusters.

Therefore, LSS-GCSS is a parameter-free AHC algorithm that does allow to automatically estimate the real number of clusters and obtain a successful final clustering solution in the context of a wide variety of clustering scenarios that may contain clusters of distinct nature and characteristics. In addition, LSS-GCSS proves to be able to perform such a task without involving any drastic increase

of the computational requirements in comparison with previous AHC methods (see section 4.4 for further details).

It is worth noticing that the present discussion directly refers to and affirmatively responds the first three research questions posed in the present thesis, as well as it addresses the falsifiability of the first research hypothesis the present thesis is based on (see section 1.3 for further details), which is validated by means of the clustering results the LSS-GCSS algorithm provide in the experimental study performed in the present chapter.

Finally, since LSS-GCSS proves to be versatile enough to face clustering problems of diverse nature without requiring any user intervention and any prior knowledge about the scenario, both the fourth research question and the second research hypothesis posed in the present thesis are next addressed in Chapter 6, where learners' activity in online discussion forums is modelled by applying the LSS-GCSS algorithm in the context of a subspace clustering-based analysis strategy.

Chapter 6

A novel strategy to model learners' activity in online discussion forums

The issue of modelling learners' activity in online discussion forums leads to a highly context-dependent analysis scenario. In addition, user intervention is habitually required in clustering-based approaches to that matter in order to provide a final clustering solution, which can easily introduce biases into the final modelling. Thus, having proved its versatility and reliability to face clustering problems of diverse nature without requiring any user intervention, LSS-GCSS algorithm seems to be a good candidate to try to minimise such biases. Therefore, with the aim of improving the analysis conditions of the previous approaches to the issue, this chapter is focused on presenting the last contributions of the present thesis, which essentially comprise the definition of a two-stage subspace clustering-based analysis strategy that, in combination with the LSS-GCSS algorithm, both be easily adaptable to the conditions of this modelling scenario and limit user's subjective intervention to the interpretation stages of the analysis process.

Consequently, this sixth chapter is structured as follows. The contributions provided in the present thesis regarding the matter of modelling learners' activity in online discussion forums from a clustering-based approach are defined in 6.1. Next, the proposed analysis strategy is detailed in section 6.2. Once defined, the proposed analysis strategy is applied in the context of a particular teaching-learning environment. Hence, a description of the input data available in such scenario is performed in section 6.3 and the implementation of the proposed analysis strategy in the defined scenario is included in sections 6.4 and 6.5. Finally, conclusions and considerations about the performed study are detailed in section 6.6, which includes a discussion directly referred to both the fourth research question and the second research hypothesis posed in section 1.3.

6.1 On modelling learners' activity in online discussion forums from a clustering perspective

As aforementioned in Chapter 1, the problem of modelling learners' activity in online discussion forums can be easily posed from the DM field as a clustering scenario, where learners with similar participation profiles are grouped together and the analysis of the resulting clusters leads to the identification of different learning behaviours. However, the different learning behaviours performed by learners in the asynchronous discussions depend on many variables (*e.g.* amount of learners in the virtual classroom, course duration, field of study, kind of subject, teaching-learning strategies promoted by teacher, etc.), which leads to a highly context-dependent modelling scenario and causes the real number of clusters to be *a priori* unknown.

Consequently, such a potentially troublesome clustering scenario needs to be tackled by means of a suitable analysis strategy, which is required to be both versatile enough to be adaptable to such a changeable problem and as less biased as possible, so that user's subjective intervention is restricted to the interpretation of the clustering results in terms of the learning behaviours performed by learners (see section 1.2.3.2 for further details).

Therefore, regarding the context of a clustering-based approach to the matter of modelling learners' activity in online discussion forums, the two last contributions of the present thesis are next defined:

- Design and implementation of an analysis strategy based on the subspace clustering paradigm for the problem of modelling learners' activity in online discussion forums. Such an analysis strategy has the ability to be easily adaptable to the highly context-dependent conditions of this modelling scenario, regardless of both what knowledge about the scenario is available in advance and which specific clustering algorithm is eventually selected.

- Application of the LSS-GCSS algorithm to the problem of modelling learners' activity in online discussion forums, hence avoiding any user intervention in the clustering stage and ensuring that the estimation of the number of clusters does not depend on any subjective criteria.

Thus, the present chapter is focused on both presenting and developing these contributions. To that effect, the benefits of their application in the specific context of the online discussion forums belonging to a particular teaching-learning environment are illustrated. Finally, the following sections also allow to complement the study on the abilities of the LSS-GCSS algorithm performed in the previous chapter, by incorporating experimental results from the framework of a complex analysis strategy in a real-word clustering scenario.

6.2 Two-stage clustering-based strategy of analysis

On the one hand, the paradigm of subspace clustering (or projected clustering) arises with the aim of overcoming some of the undesired effects that appear as the dimensionality of data grows (Bellman, 1961; Beyer et al., 1999). In a nutshell, the goal of subspace clustering is to identify clusters embedded in low-dimensional subspaces of the original data space along with their own associated dimensions. To that effect, subspace clustering methods are usually based on reducing the dimensionality of data through the use of either feature extraction or feature selection techniques in order to find optimal representation spaces for the data (Gan et al., 2007). Although there exist several clustering algorithms originally designed according to the subspace clustering premises (see section 2.3.2 for further details), the principles of this paradigm can be applied in order to decompose any clustering problem into several simpler problems, hence facilitating the subsequent cluster analysis regardless of the nature of the selected clustering algorithm.

On the other hand, the matter of modelling learners' activity in online discussion forums from a clustering perspective gives rise to a complex and high-context dependent analysis scenario. Aside from depending on many variables, learners' activity in online discussion forums can be characterised by means of multiple features of different nature, each one of which, in its turn, can result from considering one or several of the many different aspects relative to participation in asynchronous discussions (see sections 1.2.1 and 1.2.2 for further details). Thus, in order to obtain proper and complete descriptions of learners' participation considering all possible angles, it may be interesting to adopt an approach that decomposes the clustering scenario into several simpler scenarios that can be analysed separately, so that each one of them is focused on characterising a single specific aspect of learners' activity in online discussion forums. Following this logic, it seems therefore plausible to build up an analysis strategy that applies the principles of subspace clustering in order to perform such a decomposition of the clustering scenario.

Hence, being inspired in the subspace clustering paradigm and as a culmination of several previous approximations to the issue (Cobo et al., 2011, 2012), the analysis strategy proposed in the present chapter decomposes the traditional KDD process (see Figure 1.1 for further details) into two different stages, as shown in Figure 6.1:

1. **Learners' activity in online discussion forums is differently characterized in several distinct subspaces on the first analysis stage**; *i.e.* different features are used depending on the subspace, so that different subspaces describe different specific aspects of the participation in the asynchronous discussions. Thus, learners with similar activity patterns are grouped together in the same cluster belonging to the ith subspace according to the particularities of the characterisation performed in that specific subspace, therefore giving rise to **as many clustering solutions as different subspaces** are comprised in the first analysis stage

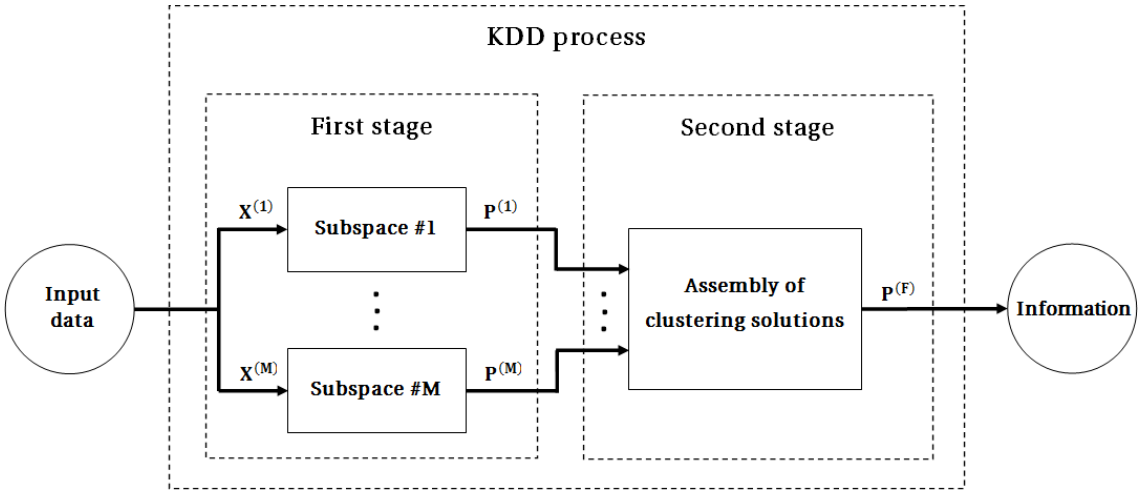


Figure 6.1: Two-stage clustering-based strategy of analysis.

$\left(\mathbf{P^{(i)}}, \forall i \in \{1, M\}\right)$. In addition, this first stage of analysis includes the evaluation and a first interpretation of every obtained clustering solution according to the particular context of the subspace it belongs to (see section 6.4 for further details).

It is worth noting that the different aspects of learners' participation characterised in this analysis strategy (*i.e.* both the number of subspaces and the specific features used in each subspace) are a direct consequence of the input data available in the scenario, which, in their turn, depend on many variables and are highly context-dependent (see section 6.3 for further details). Hence, splitting up the characterisation of the activity performed by learners into a variety of subspaces provides the proposed analysis strategy with a great deal of flexibility concerning its potential application in different contexts and scenarios. Moreover, the cluster analysis performed in the context of any subspace is always more simple and affordable than that which would take place if all the aspects of the participation were characterised in one single go, regardless of the utilised clustering algorithm.

2. **The second stage of the proposed analysis strategy essentially comprises an interpretation process** of the clustering solutions resulting from the first stage, which consists in assembling the clustering solutions obtained in the first stage and **identifying those learners belonging to the same clusters in all the subspaces**. Since each cluster has been previously characterised and interpreted in the context of the particular subspace it belongs to, it is at this point trivial to group together learners whose activity patterns have been described in the same terms on every subspace. Therefore, a final set of clusters $\left(\mathbf{P^{(F)}}\right)$ is eventually obtained, where each cluster results from the combination of specific activity patterns previously identified in the first stage of analysis. In this way, the nature of the final clusters can be interpreted in terms of complex participation profiles, so that different models of learning behaviour can be identified with different final clusters (see section 6.5 for further details).

   It is worth noting that the whole interpretation/identification process requires the user to make use of conceptual analysis tools provided by the different theoretical approaches that

describe and analyse the nature of the asynchronous discussions performed by learners in the context of an online teaching-learning environment (see section 1.2.2 for further details).

Finally, the specific implementation of the proposed analysis strategy performed in the present chapter includes the use of the LSS-GCSS algorithm on all the subspaces present in the first stage of the analysis process (see section 6.4 for further details). The main reason for this choice is that LSS-GCSS has already proved its versatility and reliability in the face of a great variety of clustering scenarios, being able both to successfully estimate the real number of clusters and to provide clustering solutions of high quality in total absence of prior knowledge about the scenario (see Chapter 5 for further details). Moreover, LSS-GCSS is a parameter-free algorithm, so that it avoids the user from intervening in the cluster analysis stage and, therefore, it favours obtaining unbiased clustering results.

It is, however, worth noting that the proposed analysis strategy is not only not linked to the use of any specific clustering method or algorithm, but it even allows to utilise a different clustering algorithm on each subspace in case it is convenient.

6.3 Input data

With the aim of illustrating its benefits, the application of the analysis strategy presented in the previous section to the modelling of the activity performed by learners in the online discussion forums belonging to a particular teaching-learning environment is studied. Specifically, the study involves three complete semesters (from February 2009 to July 2010) of three different courses belonging to the online Bachelor's Degree in Telecommunication Technologies from the Open University of Catalonia (UOC)[1]: Mathematics, Linear Systems and Circuits Theory.

In general terms, courses were taught in UOC's asynchronous web-based teaching-learning environment, every course comprised two different virtual classrooms per semester and the participation of learners in the online discussion forums of their respective classrooms was never mandatory, but always strongly recommended. Thus, the whole study involves a total amount of 672 learners distributed in eighteen different virtual classrooms and a total amount of 3842 written posts. The total academic performance rate (APR) considering the three courses and the three entire semesters indicates that 52.2% of learners pass their respective courses at the end of the semester. A more detailed description of these data is provided in Table 6.1:

Thus, data relative to the asynchronous discussions performed by learners are stored in a relational database by the online discussion forums tool included in the virtual classrooms of UOC's teaching-learning environment. Essentially, any given entry of the forums database comprises the

[1] Open University of Catalonia's website: http://www.uoc.edu/portal/en/index.html

			Posts	Learners	APR
TOTAL			**3842**	**672**	**52.2%**
First semester (107 days)	Mathematics	Classroom #1	337	83	34.9%
		Classroom #2	196	25	68%
	Linear Systems	Classroom #1	161	34	64.7%
		Classroom #2	101	14	50%
	Circuits Theory	Classroom #1	187	41	63.4%
		Classroom #2	101	13	76.9%
Second semester (119 days)	Mathematics	Classroom #1	317	87	46%
		Classroom #2	111	28	42.9%
	Linear Systems	Classroom #1	179	31	64.5%
		Classroom #2	134	20	70%
	Circuits Theory	Classroom #1	517	55	65.5%
		Classroom #2	114	15	53.3%
Third semester (107 days)	Mathematics	Classroom #1	183	58	53.4%
		Classroom #2	164	30	36.7%
	Linear Systems	Classroom #1	393	48	58.3%
		Classroom #2	69	21	28.6%
	Circuits Theory	Classroom #1	389	43	51.2%
		Classroom #2	189	26	46.2%

Table 6.1: General description of the input data. **Posts**, **Learners** and **APR** columns indicate, respectively, the total amount of written posts, the number of learners and the academic performance rate (*i.e.* percentage of learners that pass the course) per classroom, course and semester. **TOTAL** row indicates global values considering the totality of classrooms, courses and semesters.

following data: what **learner** performs what **action** at what **instant of time**. Thus, three different kinds of action are stored in the forums database:

- **SEND**: Indicates that a starting post[2] has been written by some learner. Numeric identification codes associated both to the author of the starting post and to the starting post itself are also indicated, as well as the instant of time the writing action has been performed.
- **REPLY**: Indicates that a reply post[3] has been written by some learner. Numeric identification codes associated to the author of the reply post, to the reply post itself, to the author of the replied post and to the replied post itself are also indicated, as well as the instant of time the replying action has been performed.

[2] By "starting post" it is meant that post which is written outside the context of any existing thread of conversation, so that it may give rise to a new thread of conversation.

[3] By "reply post" it is meant that post which is written inside the context of any existing thread of conversation, so that it replies to another previously written post.

- **READ**: Indicates that a post has been read for the first time by some learner. Numeric identification codes associated to the author of the reading action, to the author of the read post and to the read post itself are also indicated, as well as the instant of time the reading action has been performed.

The instant of time every action has been performed at is indicated in a date form (*dd/mm/yyyy hh:mm:ss*). Moreover, learners are identified by means of an arbitrary numeric identification code, so that their anonymity is guaranteed in the context of the forums database.

Thus, regarding the different levels of participation and units of analysis that can be considered on conceptualising what online learner participation in discussion forums is and how it can be modelled (Hrastinski, 2008), the data available in UOC's forums database does not allow to work beyond a quantitative conception of participation in terms of written posts, first time read posts, replied classmates, and pace of both writing and first time readings (see section 1.2.1 for further details). In this sense, the data stored in the forums database in UOC's teaching-learning environment is minimum, both in quantitative and qualitative terms, in comparison to the data managed in many works present in the literature, which, by way of example, include indicators relative to the total number of readings of every post performed by every learner (Calvani et al., 2010), the content of posts (Kim et al., 2011) or how learners score posts written by their classmates (Romero et al., 2013).

Finally, it is worth noting some interesting remarks concerning how the forums database has been exploded and how the obtained data have been preprocessed in the present study:

- Self-readings (*i.e.* reading actions where both the author of the action and the author of the read post are the same learner) have been dismissed, since learners get their own posts marked as read as soon as they just write them. Hence, self-readings do not provide any relevant information.

- Activity relative to teachers (*i.e.* teachers' writing and reading actions) has not been exploded, since the present study is focused on modelling learners activity only. Nonetheless, both replying and first reading actions performed by learners on teachers' posts do have been considered.

- Regarding the filtering of the exploded data, sporadic duplicated first readings of the same post performed by the same learner have been eliminated, since they are anomalies caused by possible a malfunction of the forums databased marking system.

  In addition, the typical self-introduction posts written by learners at the beginning of the semester have also been eliminated after confirming that never give rise to any conversation thread or any further interaction among learners. In this way, learners that only write

this self-introduction post throughout the whole semester are more easily identified as non-writers, which they are.

6.4 First stage of analysis

Considering the existing theoretical approaches to the conceptualisation of participation in asynchronous discussions (see section 1.2.2 for further details) and according to the available input data, the first stage of the analysis strategy is arranged in the present study as shown in Figure 6.2:

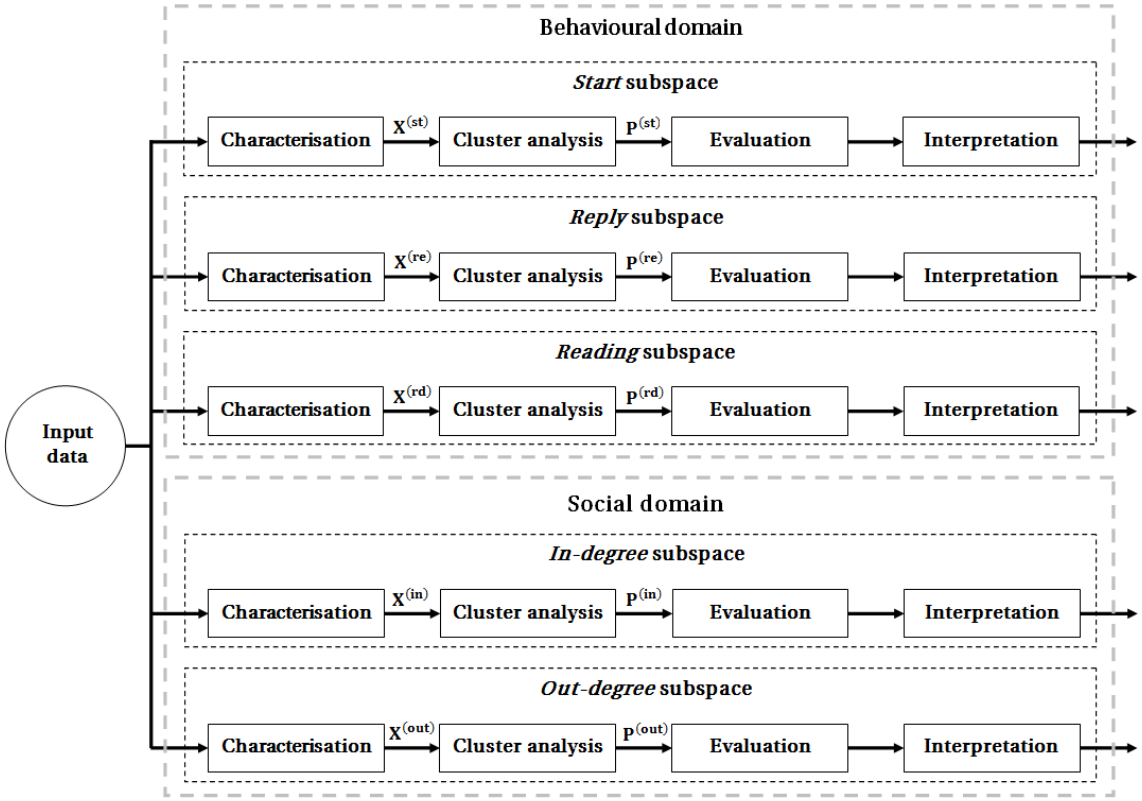


Figure 6.2: First stage of analysis.

Hence, the subspaces defined in this first stage are grouped into different domains, which represent different high-level conceptual approaches to the issue of characterising the activity performed by learners in online discussion forums. Specifically, two distinct characterisation domains of the activity performed by learners are defined in the present study, each one of which comprising different subspaces:

- **Behavioural domain**. It comprises three different subspaces where learners' activity is characterised according to strictly behaviourist conceptions of participation in asynchronous discussions (see section 1.2.2.1 for further details):
  - The *Start* subspace comprises features relative to the writing activity of starting posts (see section 6.4.1 for further details).

  - The *Reply* subspace comprises features relative to the writing activity of reply posts (see section 6.4.2 for further details).
  - The *Reading* subspace comprises features relative to the reading activity (see section 6.4.3 for further details).

- **Social domain**. It comprises two different subspaces where learners' activity is characterised according to strictly social conceptions of participation in asynchronous discussions (see section 1.2.2.2 for further details):
  - The *In-degree* subspace comprises features relative to the way learners are contacted by their classmates (see section 6.4.4 for further details).
  - The *Out-degree* subspace comprises features relative to the way learners contact with their classmates (see section 6.4.5 for further details).

As shown in Figure 6.2, every subspace includes the traditional stages of the KDD process are included in the particular context of each defined subspace.

Firstly, the activity performed by all learners involved in the present study is separately characterised in the different subspaces, giving rise to as many datasets as subspaces are defined, so that the number of objects (*i.e.* learners) is the same for all the datasets $(N = 672)$. It is the particular set of features included in each subspace what differentiates subspaces from each other, since different feature selection processes are performed in the context of different subspaces, so that features belonging to a given subspace represent singular aspects of learners' participation that are exclusive from that specific subspace.

It is worth noting that all features are normalised to zero mean and unit variance prior to the cluster analysis performed in every subspace. Moreover, the Euclidean distance (see equation 2.8) is used as proximity measure in all the subspaces.

Secondly, all the cluster analyses included in this first stage of analysis are performed by the LSS-GCSS algorithm, so that both the number of clusters and the clustering solutions associated to each subspace are automatically obtained, without requiring any user intervention.

Thirdly, every obtained clustering solution is exclusively evaluated under internal validation methods, since the present study deals with a real-world clustering scenario and ground truth solutions are not therefore available. Thus, Silhouette Coefficient $(\overline{S})$, Cophenetic Correlation Coefficient $(CPCC)$ and Kruskal-Wallis statistical test (see sections 2.4.2, 2.4.4 and 2.4.3 for further details, respectively) are utilised to evaluate and characterise the obtained clustering solutions.

And fourthly, a great variety of concepts are utilised as tools in the interpretation stages in order to identify every obtained cluster with some kind of participation profile and hence provide them with useful meanings. Such interpretation tools are provided by works present in the litera-

ture that tackle the issue of modelling learners' activity in online discussion forums by adopting a theoretical-conceptual perspective, either from behaviourist, social or constructivist approaches (see sections 1.2.2.1, 1.2.2.2 and 1.2.2.3 for further details, respectively). Additionally, contributions provided by other previous works based on modelling learners' participation in asynchronous discussions by means of a clustering perspective are also considered to such a purpose (see section 1.2.3.2 for further details).

6.4.1 *Start* subspace

Characterisation

Every learner's activity in asynchronous discussions is characterised in the *Start* subspace according to the following features ($D = 2$):

- $p^{(st)}$: Number of starting posts written by the learner, normalised by the maximum number of starting posts written by some learner of that classroom.
- $d^{(st)}$: Number of days dedicated by the learner to write, at least, one starting post, normalised by the maximum number of days dedicated by some learner of that classroom to write, at least, one starting post.

Cluster analysis

As shown in Figure 6.3, three different clusters are identified by the LSS-GCSS algorithm within the *Start* subspace:

Figure 6.3: *Start* subspace: clustering solution by LSS-GCSS.

Evaluation

The result of the evaluation of the clustering solution belonging to the *Start* subspace confirms the quality of the obtained results (see Table 6.2):

- The values of the Silhouette Coefficient considering each cluster separately indicates the presence of two high-compacted and high-isolated clusters ($\mathbf{C}_1^{(st)}$ and $\mathbf{C}_2^{(st)}$), along with a larger

	$\mathbf{C}_1^{(st)}$	$\mathbf{C}_2^{(st)}$	$\mathbf{C}_3^{(st)}$
$\overline{\mathbf{S}}$	1	0.98	0.173
CPCC	1	0.934	0.785

Table 6.2: Validity measures ($\overline{S}$ and $CPCC$) for the clustering solution shown in Figure 6.3.

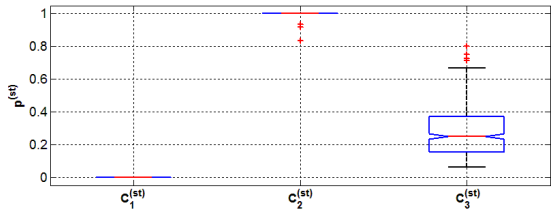
cluster of much lower density ($\mathbf{C}_3^{(st)}$). In addition, the global value of the Silhouette Coefficient ($\overline{S} = 0.751$) confirms the overall quality of the obtained clustering solution.

- The values of $CPCC$ are high enough to confirm the suitability of the representation provided by the three obtained dendrograms.

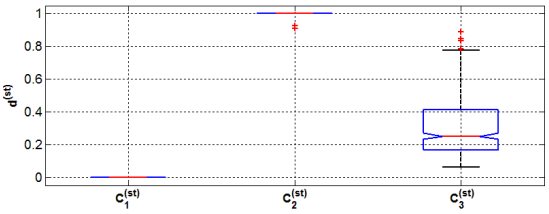
- Finally, the results of the Kruskal-Wallis test confirm that all the obtained clusters identify different statistical distributions of data, since there are significant differences ($p < 0.01$) between all clusters regarding every feature of the dataset.

Interpretation

Figure 6.4 shows the ranges of values adopted by the obtained clusters along the two features belonging to the *Start* subspace. In addition, both the amount of learners and the APR present in every cluster are detailed in Table 6.3.



(a) Boxplots corresponding to the $p^{(st)}$ feature.



(b) Boxplots corresponding to the $d^{(st)}$ feature.

Figure 6.4: Location of clusters belonging to the *Start* subspace.

	$\mathbf{C}_1^{(st)}$	$\mathbf{C}_2^{(st)}$	$\mathbf{C}_3^{(st)}$
%N	66.8%	3.1%	30.1%
APR	40.5%	95.2%	73.8%

Table 6.3: Characterisation of clusters in the *Start* subspace regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).

Thus, the clustering results obtained in the *Start* subspace allow to identify the following participation profiles:

- **Non-initiators** $\left(\mathbf{C}_1^{(st)}\right)$: Learners who present a total absence of activity regarding the writing of starting posts.

- **Leading initiators** $\left(\mathbf{C}_2^{(st)}\right)$: Learners who intensively write high amounts of starting posts throughout the entire semester, leading this particular aspect of the participation in the discussion boards of their respective classrooms.
- **Mid-class initiators** $\left(\mathbf{C}_3^{(st)}\right)$: Learners who dedicate between small and medium amounts of time to write between small and medium amounts of starting posts.

6.4.2 *Reply* subspace

Characterisation

Every learner's activity in asynchronous discussions is characterised in the *Reply* subspace according to the following features ($D = 2$):

- $p^{(re)}$: Number of reply posts written by the learner, normalised by the maximum number of reply posts written by some learner of that classroom.
- $d^{(re)}$: Number of days dedicated by the learner to write, at least, one reply post, normalised by the maximum number of days dedicated by some learner of that classroom to write, at least, one reply post.

Cluster analysis

As shown in Figure 6.5, three different clusters are identified by the LSS-GCSS algorithm within the *Reply* subspace:

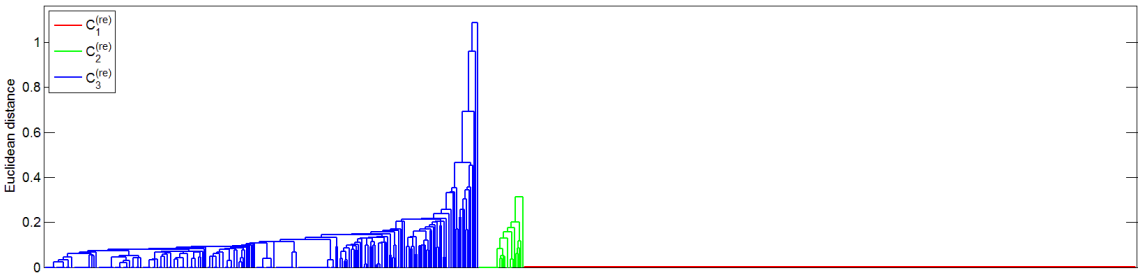


Figure 6.5: *Reply* subspace: clustering solution by LSS-GCSS.

Evaluation

The result of the evaluation of the clustering solution belonging to the *Reply* subspace confirms the quality of the obtained results (see Table 6.4):

- The values of the Silhouette Coefficient considering each cluster separately indicates the presence of two high-compacted and high-isolated clusters ($\mathbf{C}_1^{(re)}$ and $\mathbf{C}_2^{(re)}$), along with a larger

	$\mathbf{C}_1^{(re)}$	$\mathbf{C}_2^{(re)}$	$\mathbf{C}_3^{(re)}$
$\overline{\mathbf{S}}$	1	0.949	0.061
CPCC	1	0.887	0.692

Table 6.4: Validity measures ($\overline{S}$ and $CPCC$) for the clustering solution shown in Figure 6.5.

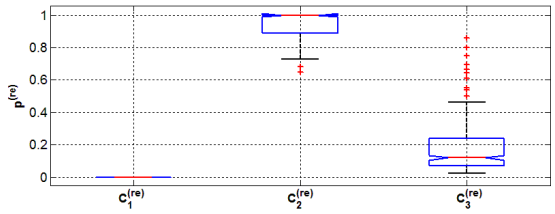
cluster of much lower density ($\mathbf{C}_3^{(re)}$). In addition, the global value of the Silhouette Coefficient $\left(\overline{S} = 0.576\right)$ confirms the overall quality of the obtained clustering solution.

- The values of $CPCC$ are high enough to confirm the suitability of the representation provided by the three obtained dendrograms.

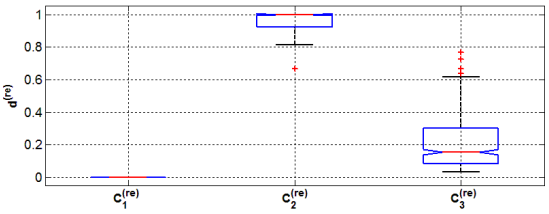
- Finally, the results of the Kruskal-Wallis test confirm that all the obtained clusters identify different statistical distributions of data, since there are significant differences $(p < 0.01)$ between all clusters regarding every feature of the dataset.

Interpretation

Figure 6.6 shows the ranges of values adopted by the obtained clusters along the two features belonging to the *Reply* subspace. In addition, both the amount of learners and the APR present in every cluster are detailed in Table 6.5.



(a) Boxplots corresponding to the $p^{(re)}$ feature.



(b) Boxplots corresponding to the $d^{(re)}$ feature.

Figure 6.6: Location of clusters belonging to the *Reply* subspace.

	$\mathbf{C}_1^{(re)}$	$\mathbf{C}_2^{(re)}$	$\mathbf{C}_3^{(re)}$
%N	56.1%	4.2%	39.7%
APR	38.2%	100%	67%

Table 6.5: Characterisation of clusters in the *Reply* subspace regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).

Thus, the clustering results obtained in the *Reply* subspace allow to identify the following participation profiles:

- **Non-repliers** $\left(\mathbf{C}_1^{(re)}\right)$: Learners who present a total absence of activity regarding the writing of reply posts.

- **Leading repliers** $\left(\mathbf{C}_2^{(re)}\right)$: Learners who intensively write high amounts of reply posts throughout the entire semester, leading this particular aspect of the participation in the discussion boards of their respective classrooms.
- **Mid-class repliers** $\left(\mathbf{C}_3^{(re)}\right)$: Learners who dedicate between small and medium amounts of time to write between small and medium amounts of reply posts.

6.4.3 *Reading* subspace

Characterisation

Every learner's activity in asynchronous discussions is characterised in the *Reading* subspace according to the following features $(D = 2)$:

- $p^{(rd)}$: Number of posts read by the learner, normalised by the maximum number of posts read by some learner of that classroom.
- $d^{(rd)}$: Number of days dedicated by the learner to the first reading of, at least, one post, normalised by the maximum number of days dedicated by some learner of that classroom to the first reading of, at least, one post.

Cluster analysis

As shown in Figure 6.7, five different clusters are identified by the LSS-GCSS algorithm within the *Reading* subspace:

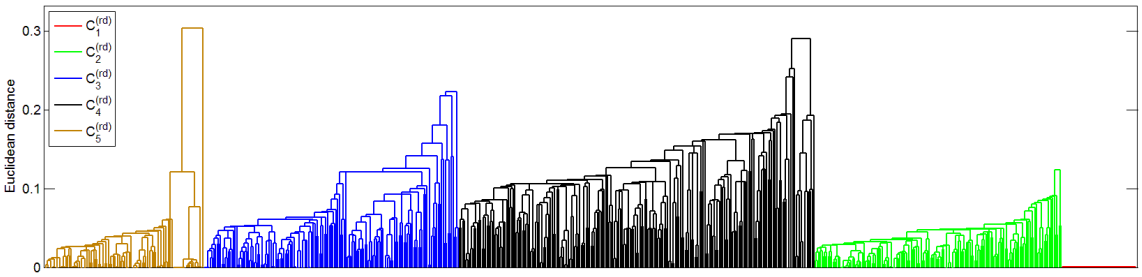


Figure 6.7: *Reading* subspace: clustering solution by LSS-GCSS.

Evaluation

The result of the evaluation of the clustering solution belonging to the *Reading* subspace confirms the quality of the obtained results (see Table 6.6):

- The values of the Silhouette Coefficient considering each cluster separately indicates the presence of two high-compacted and high-isolated clusters ($\mathbf{C}_1^{(rd)}$ and $\mathbf{C}_5^{(rd)}$), along with a moderately compacted cluster ($\mathbf{C}_3^{(rd)}$) and two other clusters of much lower density ($\mathbf{C}_2^{(rd)}$ and

	$\mathbf{C}_1^{(rd)}$	$\mathbf{C}_2^{(rd)}$	$\mathbf{C}_3^{(rd)}$	$\mathbf{C}_4^{(rd)}$	$\mathbf{C}_5^{(rd)}$
$\overline{\mathbf{S}}$	1	0.097	0.252	0.078	0.829
CPCC	1	0.484	0.518	0.663	0.572

Table 6.6: Validity measures ($\overline{S}$ and $CPCC$) for the clustering solution shown in Figure 6.7.

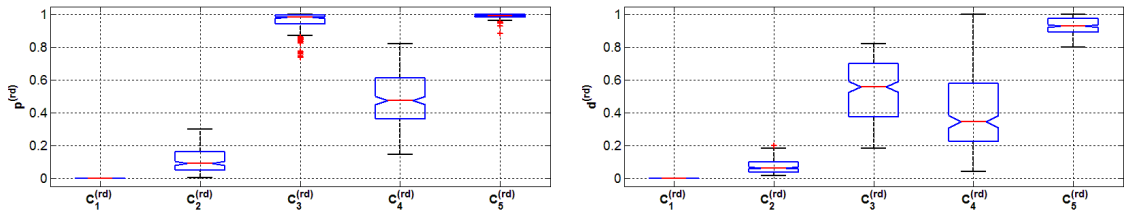
$\mathbf{C}_4^{(rd)}$). Due to the presence of a higher number of low-compacted clusters, the global Silhouette Coefficient $\left(\overline{S} = 0.296\right)$ decreases its value in comparison with the previous subspaces, but it still confirms the overall quality of the obtained clustering solution.

- The values of $CPCC$ are high enough to confirm the suitability of the representation provided by the three obtained dendrograms.

- Finally, the results of the Kruskal-Wallis test confirm that all the obtained clusters identify different statistical distributions of data, since there are significant differences $(p < 0.01)$ between all clusters regarding every feature of the dataset.

Interpretation

Figure 6.8 shows the ranges of values adopted by the obtained clusters along the two features belonging to the *Reading* subspace. In addition, both the amount of learners and the APR present in every cluster are detailed in Table 6.7.



(a) Boxplots corresponding to the $p^{(rd)}$ feature. ***(b)*** Boxplots corresponding to the $d^{(rd)}$ feature.

Figure 6.8: Location of clusters belonging to the *Reading* subspace.

	$\mathbf{C}_1^{(rd)}$	$\mathbf{C}_2^{(rd)}$	$\mathbf{C}_3^{(rd)}$	$\mathbf{C}_4^{(rd)}$	$\mathbf{C}_5^{(rd)}$
%**N**	6.8%	22.6%	23.4%	32.6%	14.6%
APR	2.2%	29.6%	65%	51.1%	92.9%

Table 6.7: Characterisation of clusters in the *Reading* subspace regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).

Thus, the clustering results obtained in the *Reading* subspace allow to identify the following participation profiles:

- **Non-readers** $\left(\mathbf{C}_1^{(rd)}\right)$: Learners who present a total absence of reading activity.

- **Low-readers** $\left(\mathbf{C}_2^{(rd)}\right)$: Learners who dedicate a small amount of time to perform first readings of a small amount of posts.

- **Intense readers** $\left(\mathbf{C}_3^{(rd)}\right)$: Learners who dedicate medium amounts of time to perform first readings of high amounts of posts.

- **Mid-class readers** $\left(\mathbf{C}_4^{(rd)}\right)$: Learners who dedicate between small and high amounts of time to perform first readings of a medium amount of posts.

- **Leading readers** $\left(\mathbf{C}_5^{(rd)}\right)$: Learners who intensively perform first readings of high amounts of posts throughout the entire semester, leading this particular aspect of the participation in the discussion boards of their respective classrooms.

6.4.4 *In-degree* subspace

Characterisation

Every learner's activity in asynchronous discussions is characterised in the *In-degree* subspace according to the following features $(D = 2)$:

- $lrd^{(in)}$: Number of learners of that classroom that have read, at least, one post written by the learner, normalised by the maximum number of learners of that classroom that have read, at least, one post written by some learner of that classroom.

- $lre^{(in)}$: Number of learners of that classroom that have replied to, at least, one post written by the learner, normalised by the maximum number of learners of that classroom that have replied to, at least, one post written by some learner of that classroom.

Cluster analysis

As shown in Figure 6.9, four different clusters are identified by the LSS-GCSS algorithm within the *In-degree* subspace:

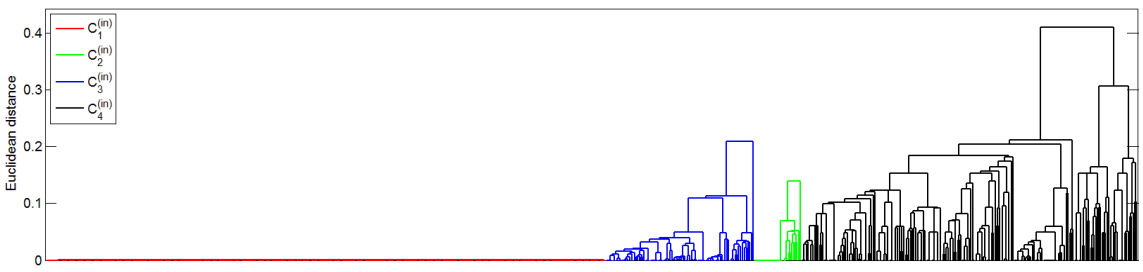


Figure 6.9: *In-degree* subspace: clustering solution by LSS-GCSS.

Evaluation

The result of the evaluation of the clustering solution belonging to the *In-degree* subspace confirms the quality of the obtained results (see Table 6.8):

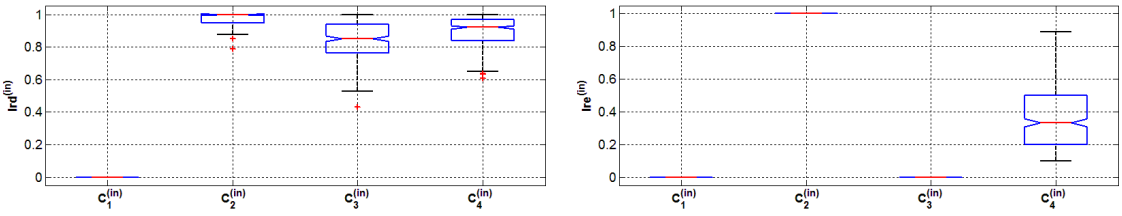
	$\mathbf{C}_1^{(in)}$	$\mathbf{C}_2^{(in)}$	$\mathbf{C}_3^{(in)}$	$\mathbf{C}_4^{(in)}$
$\overline{\mathbf{S}}$	1	0.95	0.756	0.117
CPCC	1	0.865	0.741	0.72

Table 6.8: Validity measures ($\overline{S}$ and $CPCC$) for the clustering solution shown in Figure 6.9.

- The values of the Silhouette Coefficient considering each cluster separately indicates the presence of three high-compacted and high-isolated clusters ($\mathbf{C}_1^{(in)}$, $\mathbf{C}_2^{(in)}$ and $\mathbf{C}_3^{(in)}$), along with a larger cluster of much lower density ($\mathbf{C}_4^{(in)}$). In addition, the global value of the Silhouette Coefficient $\left(\overline{S} = 0.693\right)$ confirms the overall quality of the obtained clustering solution.
- The values of $CPCC$ are high enough to confirm the suitability of the representation provided by the three obtained dendrograms.
- Finally, the results of the Kruskal-Wallis test confirm that all the obtained clusters identify different statistical distributions of data, since there are significant differences $(p < 0.01)$ between all clusters regarding every feature of the dataset, except for clusters $\mathbf{C}_1^{(in)}$ and $\mathbf{C}_3^{(in)}$ when considering the $lre^{(in)}$ feature $(p = 1)$

Interpretation

Figure 6.10 shows the ranges of values adopted by the obtained clusters along the two features belonging to the *In-degree* subspace. In addition, both the amount of learners and the APR present in every cluster are detailed in Table 6.9.



(a) Boxplots corresponding to the $lrd^{(in)}$ feature. ***(b)*** Boxplots corresponding to the $lre^{(in)}$ feature.

Figure 6.10: Location of clusters belonging to the *In-degree* subspace.

Thus, the clustering results obtained in the *In-degree* subspace allow to identify the following participation profiles:

- **Invisible learners** $\left(\mathbf{C}_1^{(in)}\right)$: Learners who are neither read not replied by any of their classmates.

	$\mathbf{C}_1^{(in)}$	$\mathbf{C}_2^{(in)}$	$\mathbf{C}_3^{(in)}$	$\mathbf{C}_4^{(in)}$
%**N**	51.3%	4.3%	13.5%	30.8%
APR	35.1%	86.2%	70.3%	68.1%

Table 6.9: Characterisation of clusters in the *In-degree* subspace regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).

- **Popular learners** $\left(\mathbf{C}_2^{(in)}\right)$: Learners who are both read and replied by most of their classmates.
- **Overlooked learners** $\left(\mathbf{C}_3^{(in)}\right)$: Learners who are read by between a medium and high amount of their classmates, whereas they are not replied by any of their classmates.
- **Integrated learners** $\left(\mathbf{C}_4^{(in)}\right)$: Learners who are read by most of their classmates and replied by a small and high amount of their classmates.

6.4.5 *Out-degree* subspace

Characterisation

Every learner's activity in asynchronous discussions is characterised in the *Out-degree* subspace according to the following features ($D = 2$):

- $lrd^{(out)}$: Number of learners of that classroom that have been read, at least, once by the learner, normalised by the maximum number of learners of that classroom that have been read, at least, once by some learner of that classroom.
- $lre^{(out)}$: Number of learners of that classroom that have been replied to, at least, once by the learner, normalised by the maximum number of learners of that classroom that have been replied to, at least, once by some learner of that classroom.

Cluster analysis

As shown in Figure 6.11, five different clusters are identified by the LSS-GCSS algorithm within the *Out-degree* subspace:

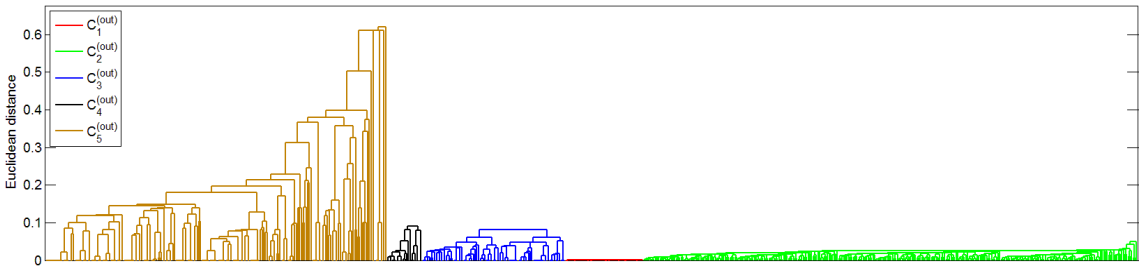


Figure 6.11: *Out-degree* subspace: clustering solution by LSS-GCSS.

Evaluation

The result of the evaluation of the clustering solution belonging to the *Out-degree* subspace confirms the quality of the obtained results (see Table 6.10):

	$\mathbf{C}_1^{(out)}$	$\mathbf{C}_2^{(out)}$	$\mathbf{C}_3^{(out)}$	$\mathbf{C}_4^{(out)}$	$\mathbf{C}_5^{(out)}$
$\overline{\mathbf{S}}$	1	0.864	0.464	0.95	0.048
CPCC	1	0.501	0.714	0.852	0.697

Table 6.10: Validity measures ($\overline{S}$ and $CPCC$) for the clustering solution shown in Figure 6.11.

- The values of the Silhouette Coefficient considering each cluster separately indicates the presence of three high-compacted and high-isolated clusters ($\mathbf{C}_1^{(out)}$, $\mathbf{C}_2^{(out)}$ and $\mathbf{C}_4^{(out)}$), along with a moderately compacted cluster ($\mathbf{C}_3^{(out)}$) and a larger clusters of much lower density ($\mathbf{C}_5^{(out)}$). In addition, the global value of the Silhouette Coefficient $\left(\overline{S} = 0.539\right)$ confirms the overall quality of the obtained clustering solution.

- The values of $CPCC$ are high enough to confirm the suitability of the representation provided by the three obtained dendrograms.

- Finally, the results of the Kruskal-Wallis test confirm that all the obtained clusters identify different statistical distributions of data, since there are significant differences $(p < 0.01)$ between all clusters regarding every feature of the dataset, except for the cases shown in Table 6.11:

	$\mathbf{C}_1^{(out)} - \mathbf{C}_2^{(out)}$	$\mathbf{C}_1^{(out)} - \mathbf{C}_3^{(out)}$	$\mathbf{C}_2^{(out)} - \mathbf{C}_3^{(out)}$	$\mathbf{C}_2^{(out)} - \mathbf{C}_4^{(out)}$	$\mathbf{C}_4^{(out)} - \mathbf{C}_5^{(out)}$
$lrd^{(out)}$				0.158	0.681
$lre^{(out)}$	1	1	1		

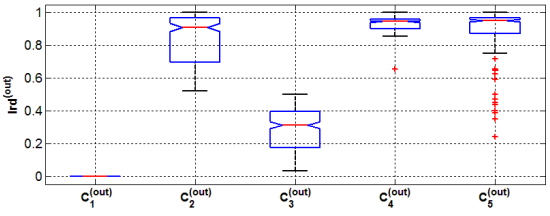
Table 6.11: Kruskal-Wallis' p-values that indicate non-significant differences in the clustering solution shown in Figure 6.11.

Interpretation

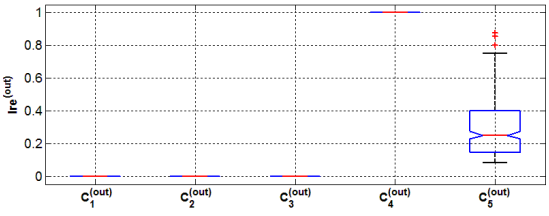
Figure 6.12 shows the ranges of values adopted by the obtained clusters along the two features belonging to the *Out-degree* subspace. In addition, both the amount of learners and the APR present in every cluster are detailed in Table 6.12.

	$\mathbf{C}_1^{(rd)}$	$\mathbf{C}_2^{(rd)}$	$\mathbf{C}_3^{(rd)}$	$\mathbf{C}_4^{(rd)}$	$\mathbf{C}_5^{(rd)}$
%N	7.1%	45.2%	13.1%	3.3%	31.3%
APR	6.3%	49%	37.5%	90.9%	69.5%

Table 6.12: Characterisation of clusters in the *Out-degree* subspace regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).



(a) Boxplots corresponding to the $lrd^{(out)}$ feature.



(b) Boxplots corresponding to the $lre^{(out)}$ feature.

Figure 6.12: Location of clusters belonging to the *Out-degree* subspace.

Thus, the clustering results obtained in the *Out-degree* subspace allow to identify the following participation profiles:

- **Introverted learners** $\left(\mathbf{C}_1^{(out)}\right)$: Learners who neither read not reply to any of their classmates.
- **Peeping learners** $\left(\mathbf{C}_2^{(out)}\right)$: Learners who, despite not replying to any of their classmates, do read between a medium and high amount of their classmates.
- **Witness learners** $\left(\mathbf{C}_3^{(out)}\right)$: Learners who, despite not replying to any of their classmates, do read between a small and medium amount of their classmates.
- **Extroverted learners** $\left(\mathbf{C}_4^{(out)}\right)$: Learners who both read and reply to most of their classmates.
- **Sociable learners** $\left(\mathbf{C}_5^{(out)}\right)$: Learners who read between a medium and high amount of their classmates and reply to between a small and high amount of their classmates.

6.5 Second stage of analysis

As a consequence of the distribution of subspaces arranged in the previous stage, the interpretation process performed in the second stage of the analysis strategy is implemented in the present study, as shown in Figure 6.13, by means of a hierarchical structure articulated through the characterisation domains initially defined in the first stage of the process. Thus, the subspace clustering solutions obtained in the first stage are not processed all at once, but the assembling process is decomposed according to the characterisation domains the subspaces are grouped into, so that subspace clustering solutions belonging to the same domain are firstly assembled each other in an initial phase and the resulting set of clusters belonging to each domain are finally assembled in the last phase of the stage.

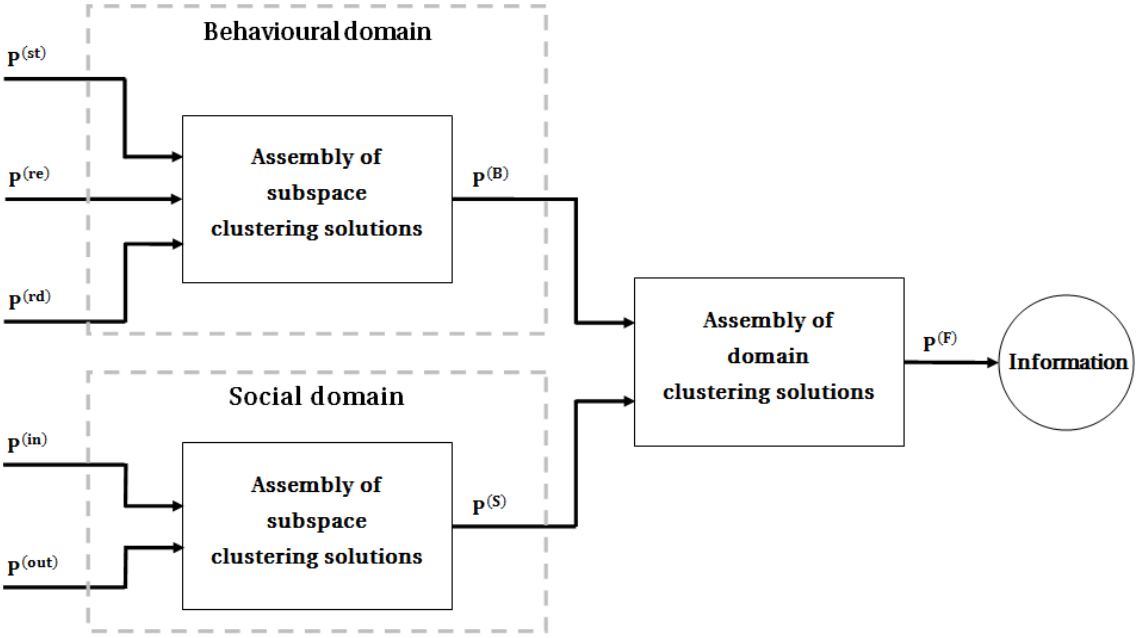


Figure 6.13: Second stage of analysis.

Such assembling processes embrace a variety of input clustering solutions and essentially consist in identifying those learners belonging to the same clusters in all the input clustering solutions, so that a new set of clusters is obtain as a result of the assembly. The models of participation associated to each of these new clusters entirely depend on the nature of the clusters of the input clustering solutions that determine their composition. Thus, depending on both the composition of the new clusters and the conceptual tools utilised in the interpretation of the results, a determinate model of learners' participation may be identified with either a single specific new cluster or the union of several new clusters that share some particular features. To that effect and similarly to the first stage of analysis, the concepts employed to interpret the results in this second stage of analysis are again provided by different theoretical approaches to the issue (see section 1.2.2 for further details), as well as by previous works that also adopt a clustering-based perspective (see section 1.2.3.2 for further details).

Therefore, learners' participation profiles are described in a first level by a set of models whose identification results from the assembly of, on the one hand, the subspace clustering solutions belonging to the Behavioural domain and, on the other hand, the subspace clustering solutions belonging to the Social domain (see sections 6.5.1 and 6.5.2 for further details, respectively). The obtained domain models –represented by the resulting sets of clusters $\mathbf{P}^{(\mathbf{B})}$ and $\mathbf{P}^{(\mathbf{S})}$– are necessarily constricted to a description of learners' participation in either purely behaviourists or purely relative to the social presence of participation terms, depending on the characterisation domain they belong to.

Finally, a final set of clusters $\left(\mathbf{P}^{(\mathbf{F})}\right)$ that give rise to the final models of learners' participation arises as a result of the assembly of the obtained domain models (see section 6.5.3 for further details). Such final models identify complex profiles of participation, whose description results

from combining interpretation elements belonging to both behaviourists and social approaches to the matter of modelling learners' activity in online discussion forums.

It is worth noticing that the actual final outcome of the analysis strategy is much richer the final set of models obtained in the last phase of this second stage, since it comprises a complex conceptual map both resulting from and representative of the entire process of analysis and it illustrates the relationships and dependencies between the variety of participation profiles and models identified throughout the different stages of the whole analysis strategy (see Figure 6.18 in section 6.5.3).

6.5.1 Behavioural domain

The assembly of the subspace clustering solutions belonging to the Behavioural domain (*i.e.* $\mathbf{P^{(st)}}$, $\mathbf{P^{(re)}}$ and $\mathbf{P^{(rd)}}$) is detailed in Table 6.13, whereas the ranges of values adopted by the resulting clusters along the features defined in the subspaces belonging to the Behavioural domain are illustrated in Figure 6.14. Moreover, both the amount of learners and the APR of the clusters corresponding to the models of participation identified in the Behavioural domain are provided in Table 6.14.

Thus, a detailed analysis of the obtained results leads to the identification of the following models of learners' participation in the Behavioural domain:

- **Shirkers** $\left(\mathbf{C}_1^{(B)} = \mathbf{C}_1\right)$: Learners who do not participate at all in the asynchronous discussions, presenting a total absence of activity regarding both writing and reading actions.
- **Lurkers** $\left(\mathbf{C}_2^{(B)} = \mathbf{C}_2 \cup \mathbf{C}_3 \cup \mathbf{C}_4 \cup \mathbf{C}_5\right)$: Learners whose participation in the asynchronous discussions is restricted to the reading of others classmates' posts, presenting a total absence of writing activity. The presence of lurkers with different levels of reading activity has been identified:
  - **Low-lurkers** $(\mathbf{C}_2)$: Lurkers who dedicate a small amount of time to perform first readings of a small amount of posts.
  - **Middle-class lurkers** $(\mathbf{C}_3)$: Lurkers who dedicate between small and high amounts of time to perform first readings of a medium amount of posts.
  - **Intense lurkers** $(\mathbf{C}_4)$: Lurkers who dedicate medium amounts of time to perform first readings of high amounts of posts.
  - **Leading lurkers** $(\mathbf{C}_5)$: Lurkers who intensively perform first readings of high amounts of posts throughout the entire semester.
- **Mere engagers** $\left(\mathbf{C}_3^{(B)} = \mathbf{C}_6 \cup \mathbf{C}_7 \cup \mathbf{C}_8 \cup \mathbf{C}_9 \cup \mathbf{C}_{10}\right)$: Learners whose participation is distinctively focused on the writing of between small and medium amounts of starting posts, presenting an absence of replying activity and a diversity of levels of reading activity.

	$\mathbf{C}_1^{(st)}$	$\mathbf{C}_2^{(st)}$	$\mathbf{C}_3^{(st)}$	$\mathbf{C}_1^{(re)}$	$\mathbf{C}_2^{(re)}$	$\mathbf{C}_3^{(re)}$	$\mathbf{C}_1^{(rd)}$	$\mathbf{C}_2^{(rd)}$	$\mathbf{C}_3^{(rd)}$	$\mathbf{C}_4^{(rd)}$	$\mathbf{C}_5^{(rd)}$	**%N**	**APR**	$\mathbf{P^{(B)}}$
$\mathbf{C}_1$	10.2%			12.2%			100%					6.8%	2.2%	$\mathbf{C}_1^{(B)}$
$\mathbf{C}_2$	28.5%			34%				84.2%				19%	28.1%	$\mathbf{C}_2^{(B)}$
$\mathbf{C}_3$	21.4%			25.5%						43.8%		14.3%	40.6%	
$\mathbf{C}_4$	13.1%			15.6%					37.6%			8.8%	52.5%	
$\mathbf{C}_5$	3.6%			4.2%							16.3%	2.4%	87.5%	
$\mathbf{C}_6$			2%	1.1%				2.6%				0.6%	50%	$\mathbf{C}_3^{(B)}$
$\mathbf{C}_7$			4%			3%		5.3%				1.2%	50%	
$\mathbf{C}_8$			8.4%	4.5%						7.8%		2.5%	64.7%	
$\mathbf{C}_9$			3.5%	1.9%					4.5%			1%	85.7%	
$\mathbf{C}_{10}$			2%	1.1%							4.1%	0.6%	100%	
$\mathbf{C}_{11}$	2.7%					4.5%		7.9%				1.8%	25%	$\mathbf{C}_4^{(B)}$
$\mathbf{C}_{12}$	9.1%					15.4%				18.7%		6.1%	53.7%	
$\mathbf{C}_{13}$	6.7%					11.2%			19.1%			4.5%	63.3%	
$\mathbf{C}_{14}$	4.2%					7.1%					19.4%	2.8%	78.9%	
$\mathbf{C}_{15}$	0.45%				7.1%						2%	0.3%	100%	
$\mathbf{C}_{16}$			29.7%			22.5%				27.4%		8.9%	60%	$\mathbf{C}_5^{(B)}$
$\mathbf{C}_{17}$			27.7%			21%			35.7%			8.3%	73.2%	
$\mathbf{C}_{18}$			16.3%			12.4%					33.7%	4.9%	97%	
$\mathbf{C}_{19}$		9.5%				0.75%				0.91%		0.3%	50%	$\mathbf{C}_6^{(B)}$
$\mathbf{C}_{20}$		9.5%				0.75%			1.3%			0.3%	100%	
$\mathbf{C}_{21}$		19%				1.5%					4.1%	0.6%	100%	
$\mathbf{C}_{22}$			0.5%		3.6%					0.46%		0.15%	100%	$\mathbf{C}_7^{(B)}$
$\mathbf{C}_{23}$			0.99%		7.1%				1.3%			0.3%	100%	
$\mathbf{C}_{24}$			5%		35.7%						10.2%	1.5%	100%	
$\mathbf{C}_{25}$		9.5%			7.1%					0.91%		0.3%	100%	$\mathbf{C}_8^{(B)}$
$\mathbf{C}_{26}$		4.8%			3.6%				0.64%			0.15%	100%	
$\mathbf{C}_{27}$		47.6%			35.7%						10.2%	1.5%	100%	

Table 6.13: Clusters in the Behavioural domain. Columns from $\mathbf{C}_1^{(st)}$ to $\mathbf{C}_5^{(rd)}$ indicate the percentage of learners in that cluster that belong to the same clusters in the other subspaces of the Behavioural domain. Columns %**N** and **APR** indicate the percentage of learners belonging to each cluster and the APR of each cluster (*i.e.* the percentage of learners belonging to that cluster that pass the course), respectively. The sum of percentages in columns $\mathbf{C}_i^{(st)}$, $\mathbf{C}_i^{(re)}$, $\mathbf{C}_i^{(rd)}$ and %**N** equals 100%.

	$\mathbf{C}_1^{(B)}$	$\mathbf{C}_2^{(B)}$	$\mathbf{C}_3^{(B)}$	$\mathbf{C}_4^{(B)}$	$\mathbf{C}_5^{(B)}$	$\mathbf{C}_6^{(B)}$	$\mathbf{C}_7^{(B)}$	$\mathbf{C}_8^{(B)}$
%**N**	6.8%	44.5%	6%	15.5%	22.2%	1.2%	1.9%	1.9%
APR	2.2%	40.1%	67.5%	58.7%	73.2%	87.5%	100%	100%

Table 6.14: Characterisation of clusters in the Behavioural domain regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).

Those mere engagers who read a small amount of posts ($\mathbf{C}_6$ and $\mathbf{C}_7$) can be distinguished from those who dedicate between medium and high amounts of time to perform first readings of between medium and high amounts of posts ($\mathbf{C}_8$, $\mathbf{C}_9$ and $\mathbf{C}_{10}$). Punctually, a marginal set of mere engagers who present negligible levels of replying activity has been also identified ($\mathbf{C}_7$).

- **Mere reactive learners** $\left(\mathbf{C}_4^{(B)} = \mathbf{C}_{11} \cup \mathbf{C}_{12} \cup \mathbf{C}_{13} \cup \mathbf{C}_{14} \cup \mathbf{C}_{15}\right)$: Learners whose participation is distinctively focused on the replying of between small and medium amounts of other classmates' posts, presenting a total absence of writing activity regarding starting posts and a diversity of levels of reading activity.

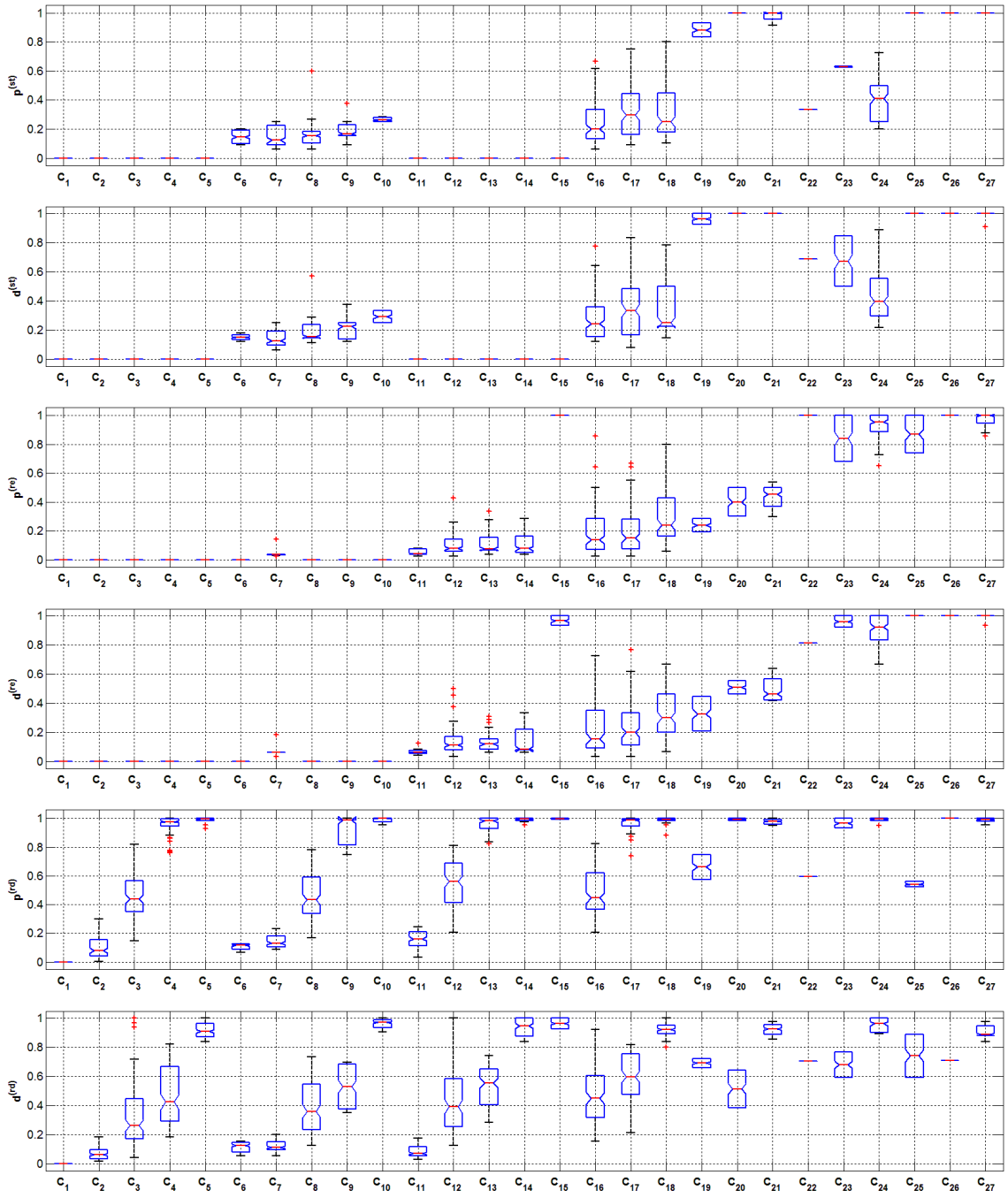


Figure 6.14: Location of clusters belonging to the Behavioural domain.

Those mere reactive learners who read a small amount of posts ($\mathbf{C}_{11}$) can be distinguished from both those who read a medium amount of posts ($\mathbf{C}_{12}$) and those who dedicate between medium and high amounts of time to perform first readings of a high amount of posts ($\mathbf{C}_{13}$, $\mathbf{C}_{14}$ and $\mathbf{C}_{15}$).

- **Mid-class workers** $\left(\mathbf{C}_5^{(B)} = \mathbf{C}_{16} \cup \mathbf{C}_{17} \cup \mathbf{C}_{18}\right)$: Learners whose participation includes the writing of between small and medium amounts of both starting and reply posts, as well as between medium and high levels of reading activity.

  Those mid-class workers who read a medium amount of posts ($\mathbf{C}_{16}$) can be distinguished from those who dedicate between medium and high amounts of time to perform first readings of a high amount of posts ($\mathbf{C}_{17}$ and $\mathbf{C}_{18}$).

- **Hard engagers** $\left(\mathbf{C}_6^{(B)} = \mathbf{C}_{19} \cup \mathbf{C}_{20} \cup \mathbf{C}_{21}\right)$: Learners whose participation is distinctively focused on the writing of high amounts of starting posts, presenting between small and medium levels of replying activity and between medium and high levels of reading activity.

  Those hard engagers who both reply a small amount of posts and read a medium amount of posts ($\mathbf{C}_{19}$) can be distinguished from those who both reply a medium amount of posts and dedicate between medium and high amounts of time to perform first readings of a high amount of posts ($\mathbf{C}_{20}$ and $\mathbf{C}_{21}$).

- **Hard reactive learners** $\left(\mathbf{C}_7^{(B)} = \mathbf{C}_{22} \cup \mathbf{C}_{23} \cup \mathbf{C}_{24}\right)$: Learners whose participation is distinctively focused on the replying of high amounts of other classmates' posts, writing a medium amount of starting posts and presenting between medium and high levels of reading activity.

  Those hard reactive learners who read a medium amount of posts ($\mathbf{C}_{22}$) can be distinguished from those who dedicate between medium and high amounts of time to perform first readings of a high amount of posts ($\mathbf{C}_{23}$ and $\mathbf{C}_{24}$).

- **Hard workers** $\left(\mathbf{C}_8^{(B)} = \mathbf{C}_{25} \cup \mathbf{C}_{26} \cup \mathbf{C}_{27}\right)$: Learners whose participation includes the writing of high amounts of both starting and reply posts, as well as between medium and high levels of reading activity.

  Those hard workers who read a medium amount of posts ($\mathbf{C}_{25}$) can be distinguished from those who dedicate between medium and high amounts of time to perform first readings of a high amount of posts ($\mathbf{C}_{26}$ and $\mathbf{C}_{27}$).

6.5.2 Social domain

The assembly of the subspace clustering solutions belonging to the Social domain (*i.e.* $\mathbf{P}^{(\mathbf{in})}$ and $\mathbf{P}^{(\mathbf{out})}$) is detailed in Table 6.15, whereas the ranges of values adopted by the resulting clusters along the features defined in the subspaces belonging to the Social domain are illustrated in Figure 6.15. Moreover, both the amount of learners and the APR of the clusters corresponding to the models of participation identified in the Social domain are provided in Table 6.16.

	$\mathbf{C}_1^{(in)}$	$\mathbf{C}_2^{(in)}$	$\mathbf{C}_3^{(in)}$	$\mathbf{C}_4^{(in)}$	$\mathbf{C}_1^{(out)}$	$\mathbf{C}_2^{(out)}$	$\mathbf{C}_3^{(out)}$	$\mathbf{C}_4^{(out)}$	$\mathbf{C}_5^{(out)}$	%**N**	**APR**	$\mathbf{P^{(S)}}$
$\mathbf{C}_1$	13.9%				100%					7.1%	6.3%	$\mathbf{C}_1^{(S)}$
$\mathbf{C}_2$	23.2%						90.9%			11.9%	35%	$\mathbf{C}_2^{(S)}$
$\mathbf{C}_3$	62.9%					71.4%				32.3%	41.5%	
$\mathbf{C}_4$			4.4%				4.5%			0.6%	75%	$\mathbf{C}_3^{(S)}$
$\mathbf{C}_5$			61.5%			18.4%				8.3%	71.4%	
$\mathbf{C}_6$				1.9%			4.5%			0.6%	50%	$\mathbf{C}_4^{(S)}$
$\mathbf{C}_7$				15%		10.2%				4.6%	61.3%	
$\mathbf{C}_8$			34.1%						14.8%	4.6%	67.7%	$\mathbf{C}_5^{(S)}$
$\mathbf{C}_9$				79.7%					78.6%	24.6%	68.5%	$\mathbf{C}_6^{(S)}$
$\mathbf{C}_{10}$		48.3%							6.7%	2.1%	85.7%	$\mathbf{C}_7^{(S)}$
$\mathbf{C}_{11}$				3.4%				31.8%		1%	100%	
$\mathbf{C}_{12}$		51.7%						68.2%		2.2%	86.7%	

Table 6.15: Clusters in the Social domain. Columns from $\mathbf{C}_1^{(in)}$ to $\mathbf{C}_5^{(out)}$ indicate the percentage of learners in that cluster that belong to the same cluster in the other subspace of the Social domain. Columns %**N** and **APR** indicate the percentage of learners belonging to each cluster and the APR of each cluster (*i.e.* the percentage of learners belonging to that cluster that pass the course), respectively. The sum of percentages in columns $\mathbf{C}_j^{(in)}$, $\mathbf{C}_j^{(out)}$ and %**N** equals 100%.

	$\mathbf{C}_1^{(S)}$	$\mathbf{C}_2^{(S)}$	$\mathbf{C}_3^{(S)}$	$\mathbf{C}_4^{(S)}$	$\mathbf{C}_5^{(S)}$	$\mathbf{C}_6^{(S)}$	$\mathbf{C}_7^{(S)}$
%**N**	7.1%	44.2%	8.9%	5.2%	4.6%	24.6%	5.4%
APR	6.3%	39.7%	71.7%	60%	67.7%	68.5%	88.9%

Table 6.16: Characterisation of clusters in the Social domain regarding their size (*i.e.* percentage of learners included in each cluster) and APR (*i.e.* percentage of learners in each cluster that pass the course).

Thus, a detailed analysis of the obtained results leads to the identification of the following models of learners' participation in the Social domain:

- **Solitaries** $\left(\mathbf{C}_1^{(S)} = \mathbf{C}_1\right)$: Learners who develop a total lack of social activity in the asynchronous discussions, avoiding both reading and replying to any of their classmates and being also neither read nor replied by any of their classmates.

- **Beholders** $\left(\mathbf{C}_2^{(S)} = \mathbf{C}_2 \cup \mathbf{C}_3\right)$: Learners who participate in the asynchronous discussions by adopting a purely observing role, which is limited to reading their classmates' posts.

  Those **mid-class beholders** who read a medium amount of their classmates ($\mathbf{C}_2$) can be distinguished from those **intense beholders** who read a high amount of their classmates ($\mathbf{C}_3$).

- **Teacher-oriented learners** $\left(\mathbf{C}_3^{(S)} = \mathbf{C}_4 \cup \mathbf{C}_5\right)$: Learners whose active interaction is strictly limited to the teacher, since, despite both reading and being read by their classmates, they do not actively engage with any other learner, which indicates that all their active participation,

either in form of starting or reply posts, is addressed to engage with and be replied by the teacher.

Those teacher-oriented learners who read between a small and medium amount of their classmates ($\mathbf{C}_4$) can be distinguished from those who read between a medium and high amount of their classmates ($\mathbf{C}_5$).

- **Detached learners** $\left(\mathbf{C}_4^{(S)} = \mathbf{C}_6 \cup \mathbf{C}_7\right)$: Learners who actively reject to engage with other learners, even when they are actually replied by their classmates. The difference between detached and teacher-oriented learners lies in the fact that the former do not react to the replies they receive from their classmates, whereas the latter are not engaged by other learners.

  Those detached learners who read between a small and medium amount of their classmates ($\mathbf{C}_6$) can be distinguished from those who read between a medium and high amount of their classmates ($\mathbf{C}_7$).

- **Ignored** $\left(\mathbf{C}_5^{(S)} = \mathbf{C}_8\right)$: Learners who are not replied by any other learner despite their active attempts to engage with them, since they reply to between a small and medium amount of their classmates.

- **Interactive learners** $\left(\mathbf{C}_6^{(S)} = \mathbf{C}_9\right)$: Learners who reach average levels of social engagement, by reading and being read by between a medium and high amount of their classmates, as well as by replying to and being replied by between a small and high amount of their classmates.

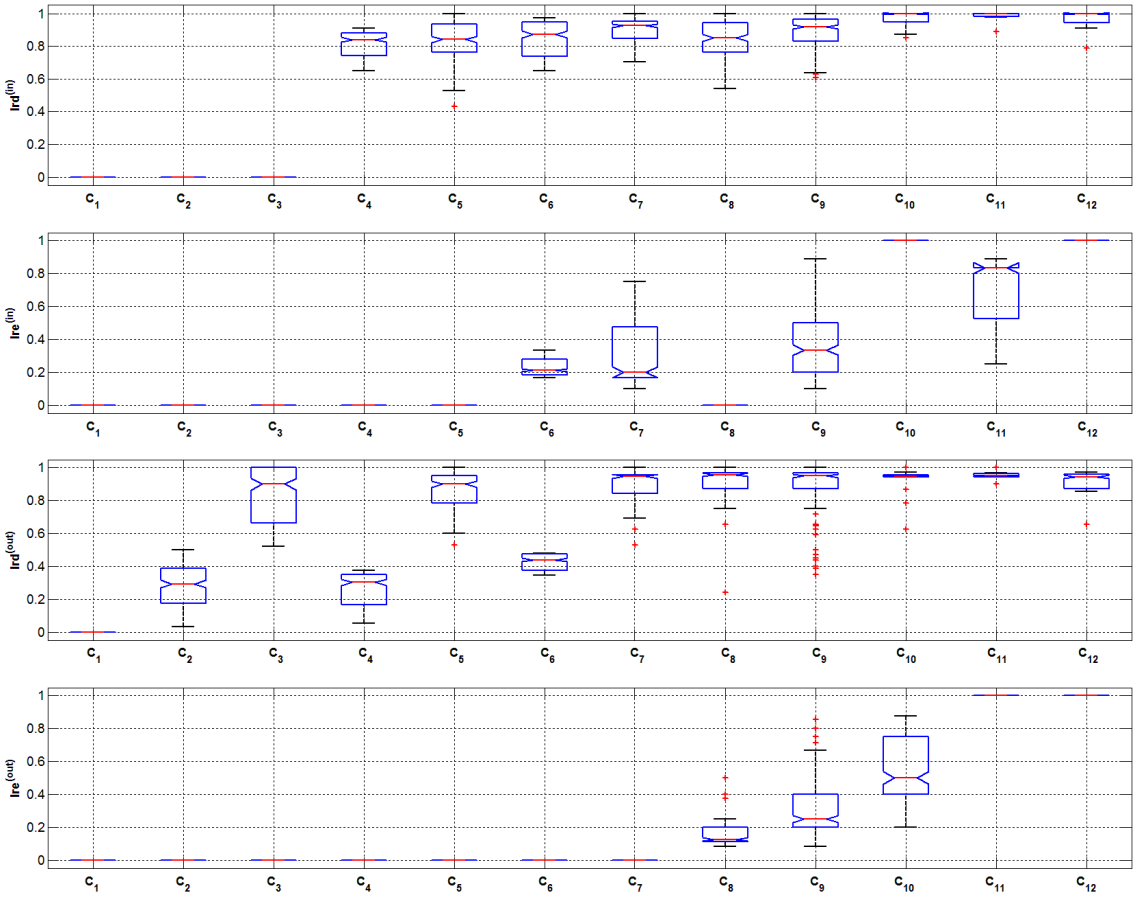


Figure 6.15: Location of clusters belonging to the Social domain.

- **Social leaders** $\left(\mathbf{C}_7^{(S)} = \mathbf{C}_{10} \cup \mathbf{C}_{11} \cup \mathbf{C}_{12}\right)$: Learners who lead the levels of social engagements in their respective classrooms, both by reading and being read by a high amount of their classmates and by replying to and being replied by between a medium and high amount of their classmates.

  Those social leaders who reply to a medium amount of their classmates $(\mathbf{C}_{10})$ can be distinguished from those who reply to a high amount of their classmates $(\mathbf{C}_{11}$ and $\mathbf{C}_{12})$.

6.5.3 Final models of learners' participation

Finally, the assembly of the sets of clusters that represent the models of participation belonging to both the Behavioural and Social domains (*i.e.* $\mathbf{P^{(B)}}$ and $\mathbf{P^{(S)}}$) is detailed in Table 6.18: columns from $\mathbf{C}_1^{(B)}$ to $\mathbf{C}_7^{(S)}$ indicate the percentage of learners in that cluster that belong to the same cluster of the other domain, and columns $\%\mathbf{N}$ and **APR** indicate the percentage of learners belonging to each cluster and the APR of each cluster (*i.e.* the percentage of learners belonging to that cluster that pass the course), respectively.

The ranges of values adopted by the resulting clusters along the features defined in the Behavioural and Social domains are illustrated in Figures 6.16 and 6.17, respectively. Moreover, both the amount of learners and the APR of the clusters corresponding to the final models of participation identified in this last phase of the second stage of analysis are provided in Table 6.17.

	$\mathbf{C}_1^{(F)}$	$\mathbf{C}_2^{(F)}$	$\mathbf{C}_3^{(F)}$	$\mathbf{C}_4^{(F)}$	$\mathbf{C}_5^{(F)}$	$\mathbf{C}_6^{(F)}$	$\mathbf{C}_7^{(F)}$
$\%\mathbf{N}$	7.1%	44.2%	6%	15.5%	21%	3.1%	3.1%
APR	6.3%	39.7%	67.5%	58.7%	73%	90.5%	95.2%

Table 6.17: Characterisation of the final models of learners' participation regarding their size (*i.e.* percentage of learners that belong to each model) and APR (*i.e.* percentage of learners belonging to each model that pass the course).

Thus, a detailed analysis of the obtained results leads to the identification of the final models of learners' participation:

- **Non-participants** $\left(\mathbf{C}_1^{(F)} = \mathbf{C}_1 \cup \mathbf{C}_2\right)$: Learners who do not participate at all in the asynchronous discussions, presenting a total absence of any kind of activity and social presence.

  Learners belonging to $(\mathbf{C}_1)$ are solitary shirkers, whereas $(\mathbf{C}_2)$ represents a marginal amount of solitary lurkers who have performed a negligible reading action of some post written by the teacher.

- **Listeners** $\left(\mathbf{C}_2^{(F)} = \mathbf{C}_3\right)$: Learners who restrict their participation to merely observing other learners' discussions, by exclusively performing lurking and beholding behaviours.

	$\mathbf{C}_1^{(B)}$	$\mathbf{C}_2^{(B)}$	$\mathbf{C}_3^{(B)}$	$\mathbf{C}_4^{(B)}$	$\mathbf{C}_5^{(B)}$	$\mathbf{C}_6^{(B)}$	$\mathbf{C}_7^{(B)}$	$\mathbf{C}_8^{(B)}$	$\mathbf{C}_1^{(S)}$	$\mathbf{C}_2^{(S)}$	$\mathbf{C}_3^{(S)}$	$\mathbf{C}_4^{(S)}$	$\mathbf{C}_5^{(S)}$	$\mathbf{C}_6^{(S)}$	$\mathbf{C}_7^{(S)}$	**%N**	**APR**	$\mathbf{P}^{(\mathbf{F})}$
$\mathbf{C}_1$	100%								95.8%							6.8%	2.2%	$\mathbf{C}_1^{(F)}$
$\mathbf{C}_2$		0.67%							4.2%							0.3%	100%	
$\mathbf{C}_3$		99.3%								100%						44.2%	39.7%	$\mathbf{C}_2^{(F)}$
$\mathbf{C}_4$			50%								33.3%					3%	70%	$\mathbf{C}_3^{(F)}$
$\mathbf{C}_5$			37.5%									42.9%				2.2%	73.3%	
$\mathbf{C}_6$			12.5%											3%		0.74%	40%	
$\mathbf{C}_7$				24%							41.7%					3.7%	64%	$\mathbf{C}_4^{(F)}$
$\mathbf{C}_8$				9.6%								28.6%				1.5%	50%	
$\mathbf{C}_9$				20.2%									67.7%			3.1%	57.1%	
$\mathbf{C}_{10}$				42.3%										26.7%		6.5%	56.8%	
$\mathbf{C}_{11}$				3.8%											11.1%	0.6%	75%	
$\mathbf{C}_{12}$					10.1%						25%					2.2%	86.7%	$\mathbf{C}_5^{(F)}$
$\mathbf{C}_{13}$					6%							25.7%				1.3%	44.4%	
$\mathbf{C}_{14}$					6.7%								32.3%			1.5%	90%	
$\mathbf{C}_{15}$					71.8%									64.8%		15.9%	72%	
$\mathbf{C}_{16}$					5.4%										22.2%	1.2%	75%	$\mathbf{C}_6^{(F)}$
$\mathbf{C}_{17}$							92.3%								33.3%	1.8%	100%	
$\mathbf{C}_{18}$							7.7%							0.61%		0.15%	100%	
$\mathbf{C}_{19}$						37.5%								1.8%		0.45%	100%	$\mathbf{C}_7^{(F)}$
$\mathbf{C}_{20}$						50%									11.1%	0.6%	75%	
$\mathbf{C}_{21}$								38.5%						3%		0.74%	100%	
$\mathbf{C}_{22}$								61.5%							22.2%	1.2%	100%	
$\mathbf{C}_{23}$						12.5%						2.9%				0.15%	100%	

Table 6.18: Clusters corresponding to the final models of learners' participation. The sum of percentages in columns $\mathbf{C}_j^{(B)}$, $\mathbf{C}_j^{(S)}$ and %**N** equals 100%.

Listeners present a passive attitude regarding the asynchronous discussions that take place in their respective classrooms and decide to stay on the edge of any possible active interaction with other learners.

- **Questioners** $\left(\mathbf{C}_3^{(F)} = \mathbf{C}_4 \cup \mathbf{C}_5 \cup \mathbf{C}_6\right)$: Learners who are distinctively focused on writing starting posts, who are read and can easily be replied, and who do not engage previously started conversations.

  Questioners are essentially mere engagers who can either develop a teacher-oriented activity $(\mathbf{C}_4)$, simply be detached from the interests of their classmates $(\mathbf{C}_5)$ or, more marginally, be slightly interactive learners $(\mathbf{C}_6)$.

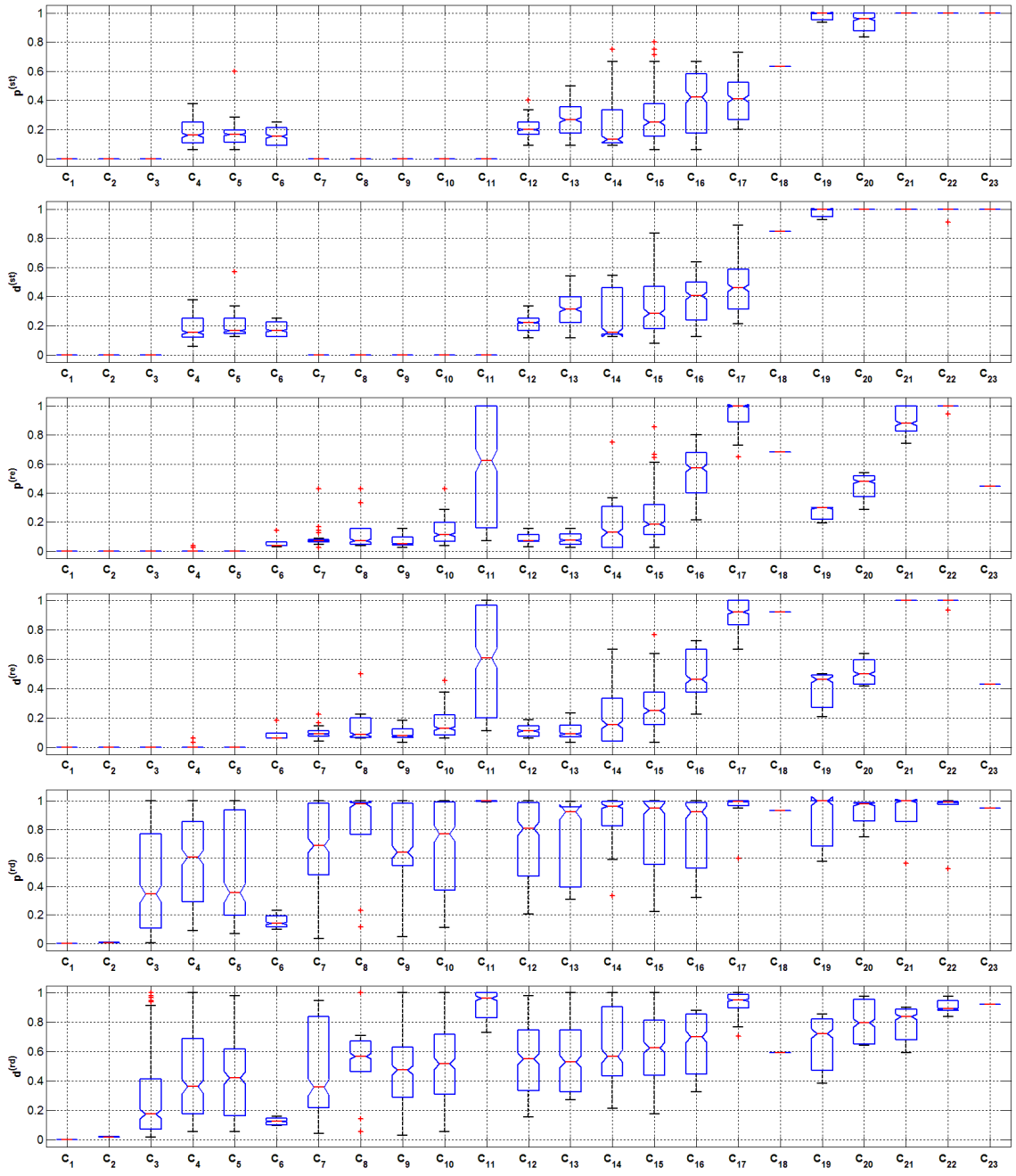


Figure 6.16: Location of clusters corresponding to the final models along the features belonging to the Behavioural domain.

- **Joining conversationalists** $\left(\mathbf{C}_4^{(F)} = \mathbf{C}_7 \cup \mathbf{C}_8 \cup \mathbf{C}_9 \cup \mathbf{C}_{10} \cup \mathbf{C}_{11}\right)$: Learners who are distinctively focused on engaging previously started conversations, who are read and can easily be replied, and who by no means start any new thread of discussion.

  Joining conversationalists are essentially mere reactive learners who can either develop a teacher-oriented behaviour detached from the rest of their classmates ($\mathbf{C}_7$ and $\mathbf{C}_8$), be ignored by the rest of their classmates ($\mathbf{C}_9$), play the role of interactive learners in originally foreign discussions ($\mathbf{C}_{10}$), or even be social leaders with between medium and high levels of activity and popularity who are exclusively focused on joining existing conversations ($\mathbf{C}_{11}$).

- **Regular participants** $\left(\mathbf{C}_5^{(F)} = \mathbf{C}_{12} \cup \mathbf{C}_{13} \cup \mathbf{C}_{14} \cup \mathbf{C}_{15}\right)$: Learners who present average levels of activity of diverse nature in overall terms.

  Regular participants are mainly mid-class workers with an interactive presence in the asynchronous discussions ($\mathbf{C}_{12}$), who can also be either teacher-oriented ($\mathbf{C}_{13}$), detached ($\mathbf{C}_{14}$) or ignored ($\mathbf{C}_{15}$) learners.

- **Dialogical learners** $\left(\mathbf{C}_6^{(F)} = \mathbf{C}_{16} \cup \mathbf{C}_{17} \cup \mathbf{C}_{18}\right)$: Learners who are both distinctively and intensely focused on engaging previously started conversations, who are specially interested in engaging dialogue with other learners, who generate high levels of interaction with their classmates, and who occasionally start new threads of discussion as well.

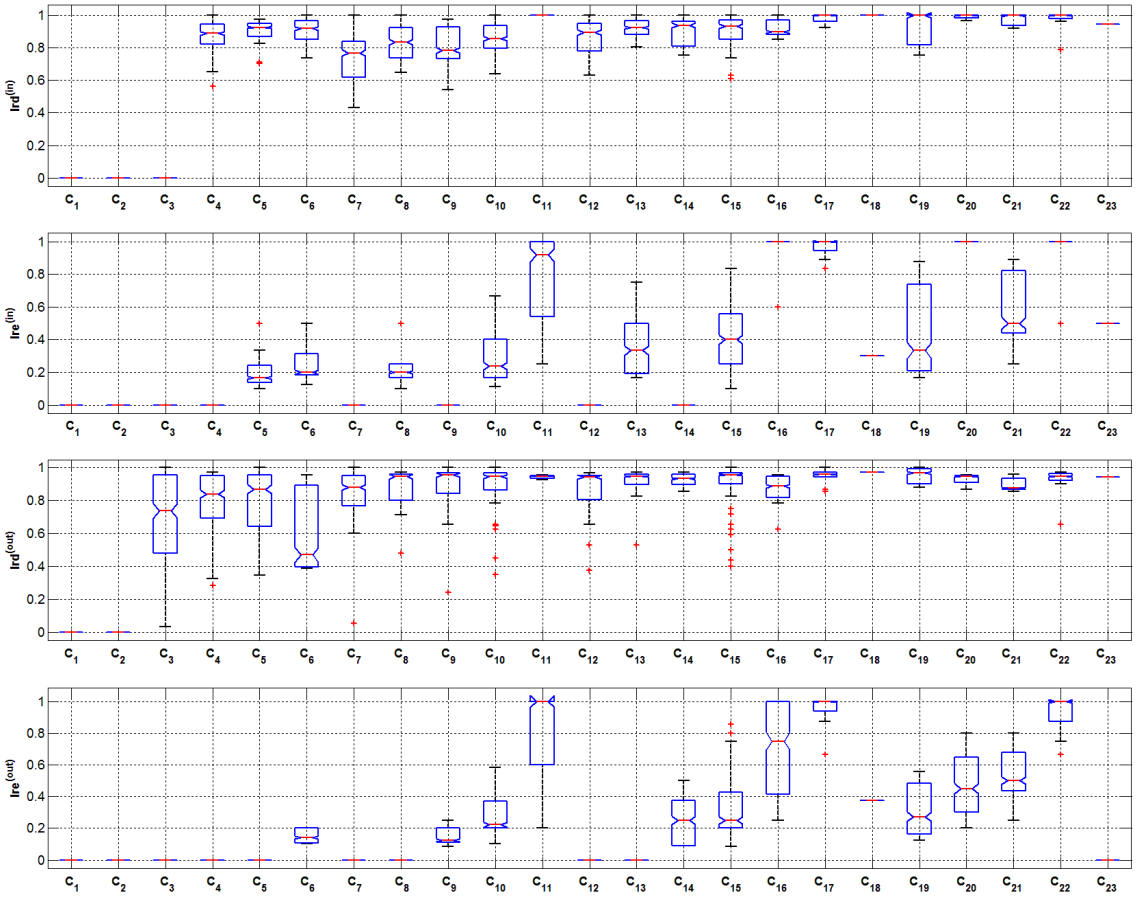


Figure 6.17: Location of clusters corresponding to the final models along the features belonging to the Social domain.

Dialogical participants are either mid-class workers ($\mathbf{C}_{16}$) or, mostly, hard-reactive learners ($\mathbf{C}_{17}$) at the top of the social engagement levels of their respective classrooms, as well as, more marginally, simply interactive learners specially focused on performing an intense participation in conversations started by other learners ($\mathbf{C}_{17}$).

- **Leading participants** $\left(\mathbf{C}_7^{(F)} = \mathbf{C}_{19} \cup \mathbf{C}_{20} \cup \mathbf{C}_{21} \cup \mathbf{C}_{22} \cup \mathbf{C}_{23}\right)$: Learners who present high levels of activity of diverse nature in overall terms.

  Leading participants can be either hard engagers ($\mathbf{C}_{19}$ and $\mathbf{C}_{20}$) or hard workers ($\mathbf{C}_{21}$ and $\mathbf{C}_{22}$) who present high levels of interaction and social engagement. It results particularly interesting the participation model represented by $\mathbf{C}_{23}$, which identifies an extremely singular leading participant who, despite of both presenting high levels of activity of all kinds and being read and replied by the majority of classmates, develops a behaviour completely detached from the rest of learners in the classroom.

Finally, the final outcome that illustrates the entirety of the analysis process implemented in the present chapter is shown in Figure 6.18, where all the participation profiles and models of behaviour identified throughout the different stages of the proposed analysis strategy are detailed and linked each other.

6.6 Discussion

Whereas most of the works that adopt a clustering-based perspective regarding the matter of modelling learners' activity in online discussion forums perform the cluster analysis on a single dataset that contain a heterogeneity of features (see section 1.2.3.2 for further details), a two-stage analysis strategy based on the subspace clustering paradigm that decompose the problem into a multiplicity of smaller and simpler clustering scenarios is proposed and developed in the present chapter.

Such analysis strategy proves to be able to not only identify and describe a certain amount of models of participation, but to provide a complex characterisation of the activity performed by learners in the asynchronous discussions materialised in the form of a conceptual map, which comprises a variety of hierarchically linked models that contextualise each other. Moreover, the particular implementation of the proposed analysis strategy in the context of a specific online teaching-learning environment shows the advantages of organising the subspaces into domains according to their similarities, which both facilitates the interpretation process of the results by decomposing it into different phases and gives directly rise to a hierarchy of models.

Additionally, combining the proposed analysis strategy with the use of the LSS-GCSS algorithm allows to automatically obtain all the subspace clustering solutions, thus limiting user's subjective

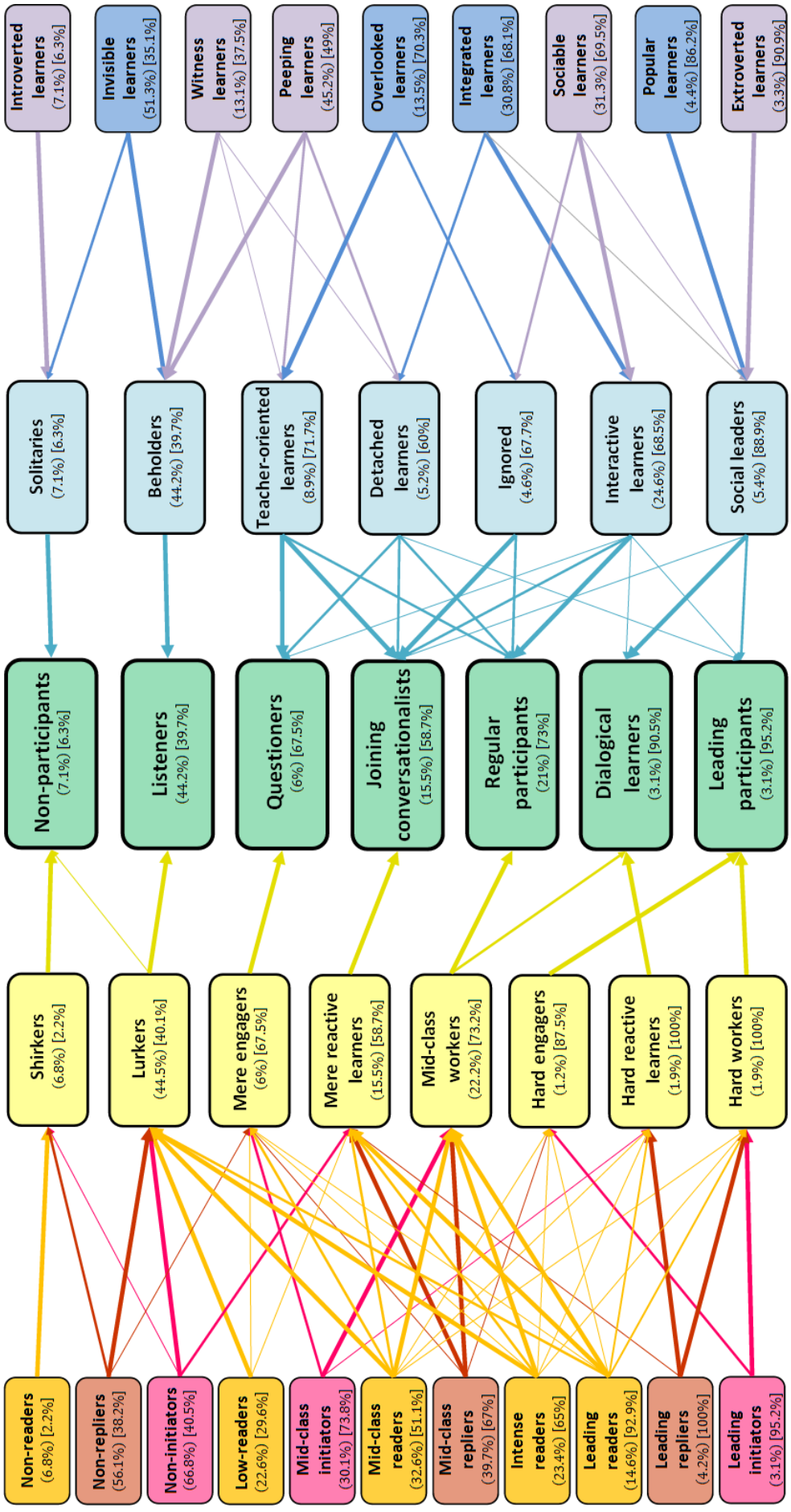


Figure 6.18: Final models of learners' participation in online discussion forums. Models belonging to the same subspace or domain are identified by their colour. Values in parentheses and brackets indicate the percentage of learners belonging to that model and the APR of that model (*i.e.* the percentage of learners belonging to that model that pass the course), respectively. The thickness of links indicates the relative amount of learners belonging to the origin model that also belongs to the destination model: the thicker the link, the higher the amount of learners transferred from the origin model to the destination model.

intervention to the interpretation stages of the analysis process and, therefore, minimising the possible biases user's prior expectations may introduce into the obtained results.

It is worth noticing that the present discussion directly refers to and affirmatively responds the fourth research question posed in the present thesis, as well as it addresses the falsifiability of the second research hypothesis the present thesis is based on (see section 1.3 for further details), which is validated by the outcomes obtained in the study performed in the present chapter.

Finally, it may also be interesting to pay heed to the fact that the results obtained in the context of the study performed in the present chapter suggest the possibility of widen the scope of application of the proposed analysis strategy beyond the strict task of modelling learners' behaviour:

- Regarding how to provide learners with a better and more personalised feedback (Ngwenya et al., 2008), or even in the context of assessment tasks (Yang, 2004), different specific strategies can be developed by teachers depending on the model of participation adopted by learners in the online discussion forums.

- The information resulting from the proposed analysis strategy can easily be utilised to feed visualisation tools present in online teaching-learning environments that show useful indicators to both teachers and learners, such as, by way of example, different aspects of participation and academic performance represented in radiant graphs (Calvani et al., 2010; Rabbany et al., 2011) (see section 1.2.3.1 for further details) or personal learning information sketched into data portraits (García-Solórzano et al., 2012).

- Inasmuch as the average academic performance of learners that develop a specific model of participation can sensibly vary between models (see Tables 6.3, 6.5, 6.7, 6.9, 6.12, 6.14, 6.16 and 6.17 for further details), it seems reasonable to think that the information relative to these models may certainly be helpful in order to predict learners' academic performance.

  Aside from providing participation features that can be correlated with the academic performance to some extent, knowing how a learner participates in the asynchronous discussions can utilised to particularise a distinctive prediction scenario depending on the model, since the balancing between classes (learners that pass the course vs. learners that fail the course) varies depending on the model of participation the learner belongs to.

  Such connection between both scopes of application (modelling and prediction) can perfectly be applied both to prediction scenarios that consider data unrelated to the asynchronous discussions (Antunes, 2011) and to try to predict final academic performance exclusively from data resulting from learners' activity in online discussion forums (Romero et al., 2013).

Chapter 7

Conclusions and further work

As aforementioned at the beginning of Chapter 1, the present thesis arises from the issue of the modelling of learners' activity in online discussion forums from a clustering-based perspective, which gives rise to a highly context-dependent analysis scenario where the real number of clusters is *a priori* unknown. Such an underlying problem is one of the best-known issues of the clustering paradigm. With the aim of avoiding any user intervention in the estimation of the number of clusters, which may easily lead to the appearance of undesired biases in the obtained models, a novel parameter-free AHC algorithm (LSS-GCSS) is proposed in the present thesis. Experimental results show that LSS-GCSS algorithm is able to provide optimal clustering solutions in the face of a great variety of clustering scenarios, both outperforming clustering algorithms most widely used in practice and involving computational requirements comparable to those of other AHC algorithms of its kind. Finally, the issue of modelling learners' participation in the online discussion forums belonging to a particular teaching-learning environment is tackled by means of LSS-GCSS algorithm, which is applied in the context of a two-stage strategy of analysis based on the subspace clustering paradigm. The combination of both techniques limits user's subjective intervention to the interpretation stages of the analysis process and lead to a complete modelling of the activity performed by learners.

In this last chapter, both the conclusions and the future lines of work derived from the present thesis are detailed. Firstly, conclusions are presented in section 7.1, following the structure of research questions and research hypotheses defined in the first chapter of this thesis. Finally, possible further work is proposed in section 7.2.

7.1 Conclusions

As a conclusion of the present thesis, the following lines address the matter of providing suitable response to the four research questions initially posed in section 1.3. Additionally, the matter of the falsifiability of the two research hypotheses directly derived from the research questions, also posed in section 1.3 and around which the present thesis has been developed, is tackled as well.

Q1. May the real number of clusters and the final clustering solution of a dataset be automatically obtained by means of an agglomerative hierarchical clustering algorithm?

On the one hand, despite not demanding the number of clusters to be identified as an input parameter, AHC basic methods require a postprocessing stage subsequent to the obtaining of the dendrogram in order to eventually identify a determinate number of clusters in the dataset, which may involve switching the estimation problem of the number of clusters for a parameter selection problem (see section 3.1.3for further details).

Thus, as a first contribution of the present thesis, the survey performed in section 2.5.1 indicates the existence of multiple different approaches (*e.g.* relative validity approach, self-refining consensus approach, model-based approach, etc.) to the issue of the estimation of the real number of clusters in a dataset. Many of these approaches make use of parameter-dependent AHC algorithms to generate a diversity of clustering solutions and, according to some kind of criterion, select one of them as a final result. Hence, despite their different inconveniences may easily make them to provide non-optimal clustering solutions, this class of approaches prove that parameter-dependent AHC algorithms –as well as other parameter-dependent clustering methods– can be utilised to try to automatically estimate the real number of clusters in a dataset.

On the other hand, there also exist parameter-free approaches to clustering that propose the implementation of parameter-free AHC algorithms specifically designed to automatically generate twofold clustering solutions (*i.e.* hard partitional solutions constructed as a result of an agglomeration process) by themselves, without involving any complementary strategy or further postprocessing stage (see 2.5.1.7 for further details). Thus, parameter-free AHC algorithms also allow to automatically obtain a final clustering solution without requiring the number of clusters –or any other input parameter– to be provided in advance.

Q2. May this agglomerative hierarchical clustering algorithm deal with datasets of different nature that can contain clusters of distinct characteristics?

Due to the significant limitations they present regarding their scope of application (their performance depends to a high extent on the characteristics of the clustering scenario), neither the dif-

ferent strategies that make use of parameter-dependent AHC methods nor the implementations of parameter-free AHC algorithms made to date can guarantee to be able to properly deal with datasets of different nature that may contain clusters of distinct characteristics (see sections 2.6 and 3.3 for further details, respectively).

In this context, the main contributions of the present thesis consist of the definition of two new cluster merging criteria and, as a direct consequence, the design and implementation of LSS-GCSS, a novel parameter-free AHC algorithm (see Chapter 4 for further details). In overall terms, the main goal of LSS-GCSS is to be able to automatically provide optimal clustering solutions in all kind of clustering scenarios, regardless of the nature of the datasets and the characteristics of the clusters they contain. Thus, with the aim of verifying whether LSS-GCSS achieves such a goal and overcomes the lacks of other AHC algorithms, the next contribution of the present thesis consists in the experimental study carried out in Chapter 5, which draws two main conclusions.

Firstly, LSS-GCSS is certainly able to handle a great variety of clustering problems (*i.e.* LSS-GCSS properly handles different kinds of data, clusters with different densities and distributions, touching and overlapped clusters, arbitrary-shaped clusters, etc.) and provide optimal clustering solutions with neither any user intervention nor any prior knowledge about the scenario required (see sections 5.2 and 5.3 for further details). Its most significant limitation lies in requiring clusters to be more separated as either the unbalancing between clusters or the number of clusters in the dataset increase (see sections 5.2.3 and 5.2.4 for further details, respectively).

And secondly, LSS-GCSS clearly proves to be more versatile and reliable than any of the clustering algorithms most widely used in practice, including both basic AHC methods and earlier parameter-free AHC algorithms. In the face of a high diversity of clustering scenarios and suffering a comparative disadvantage (since negative effects an improper parametrisation might cause on the performance of parameter-dependent algorithms have not been considered), LSS-GCSS outperforms both HPC and AHC algorithms in overall terms, being even able to provide the best clustering results in many of the scenarios (see section 5.4 for further details).

Therefore, unlike other AHC algorithms that can be employed to estimate the real number of clusters in a dataset, LSS-GCSS proves to be able to successfully perform such a task in the face of datasets of different nature that can contain clusters of distinct characteristics and without requiring any user intervention.

Q3. May this agglomerative hierarchical clustering algorithm automatically provide optimal clustering solutions without drastically increasing its computational requirements in comparison with other agglomerative hierarchical clustering algorithms?

As it is analytically proved in section 4.3, the computational requirements of the LSS-GCSS algorithm are of the same order than those of both basic AHC methods and other parameter-free AHC

algorithms; *i.e.* they remain $O(N^2)$ in storage terms and $O(N^2 \log N)$ in time terms, thus quadratically growing with the number of objects in the dataset (N).

Hence, LSS-GCSS proves to be able to provide optimal clustering solutions without drastically increasing its computational requirements in comparison with other AHC algorithms.

Q4. May this agglomerative hierarchical clustering algorithm be employed to model learners' activity in online discussion forums limiting the user intervention to the interpretation of the final results?

With the aim of improving the analysis conditions imposed by the previous approaches to the issue of modelling learners' participation in discussion forums from a clustering perspective (they require the user to participate in the cluster analysis stage in order to provide a final estimation of the number of clusters), the last contributions of the present thesis are made in Chapter 6 .

To that effect, a two-stage analysis strategy based on the subspace clustering paradigm is defined in section 6.2. Despite not being linked to the use of any specific clustering method or algorithm, the proposed analysis strategy is applied in combination with the LSS-GCSS algorithm in order to model learners' participation in the online discussion forums belonging to a particular teaching-learning environment (see sections 6.4 and 6.5 for further details).

As the obtained results indicate, LSS-GCSS proves to be able to provide a complete modelling of the activity performed by learners, limiting the user intervention to the interpretation stages of the proposed analysis strategy (see section 6.6 for further details).

H1. Agglomerative hierarchical clustering methods are suitable for automatically determining the real number of clusters and providing the clustering solution on datasets of different nature that may contain clusters of distinct characteristics.

As a consequence of the responses provided to the first three research questions, the first research hypothesis of the present thesis proves to be false, since it is only validated by means of the LSS-GCSS algorithm, but not so by basic AHC methods, nor by earlier parameter-free AHC algorithms.

H2. Learners' participation in online discussion forums can be properly modelled and described by means of a clustering-based strategy that automatically provides the clustering solution and limits any user intervention to the interpretation stage of the analysis process.

As a consequence of the response provided to the last and fourth research question, the second research hypothesis of the present thesis proves to be true, since it is validated by combined use of the proposed two-stage analysis strategy and the LSS-GCSS algorithm.

7.2 Further work

Finally, as a culmination of the present thesis, possible future lines of work are next proposed.

Regarding the cluster merging criteria (LSS and GCSS) developed in this thesis, it would interesting to study the possibility of developing equivalent merging rules in the context of other AHC methods different from the SL-based ones (*e.g.* CL, AL, Ward's, etc.), so that novel proposals of parameter-free AHC algorithms based on these new merging rules could arise.

Regarding the LSS-GCSS algorithm (and, actually, the rest of AHC methods), it would be interesting to compensate its computational requirements $\left(O(N^2)\right)$ in order to make possible its application to very large-scale datasets (*i.e.* big data applications). To that effect, it would deserve further study the viability of the two-stage clustering process next outlined. In the first stage, either a grid-based or a density-based clustering algorithm –whose computational cost is linear with respect to the number of objects in the dataset $(O(N))$– would always identify the same large number of clusters (*e.g.* $N' = 10000$), both extremely much lower than N and much greater than the real number of clusters in the dataset $(N \ggg N' \gg K_{opt})$. Thus, in the second stage, LSS-GCSS would automatically obtain the final clustering solution handling an input dataset whose size would always be N'. In this way, the total computational cost of the process would be affordable regardless of the size of the input dataset, since it would always be linear with respect to N; *i.e.* $O(N) + O\left(N'^2\right)$, where $N > N'^2$.

Regarding the issue of modelling learners' activity in online discussion forums from a clustering perspective, it would be interesting to apply the proposed two-stage analysis strategy to discussion boards that provide richer input data, so that learners' activity could be characterised by a greater variety of features, more domains and subspaces could be arranged and, therefore, more complex participation profiles could be modelled. In fact, it would also be interesting to widen the scope of the modelling scenario and apply the proposed analysis strategy to model other kinds of activity that takes place in an online teaching-learning environment; *e.g.* learners' activity performed in the context of remote practice laboratories, such as those belonging to both Bachelor's Degree in Telecommunication Technologies and Master's Degree in Telecommunications Engineering from the Open University of Catalonia (UOC).

And, finally, regarding other e-learning tasks different from the modelling of learners' behaviour, it would be interesting to explore how the outcome provided by the proposed two-stage analysis strategy (see Figure 6.18 in section 6.5.3) could be integrated with other visualisation tools or utilised to build robust predictors of learners' academic performance.

References

Adams, D. M. and Hamm, M. E. (1990). *Cooperative Learning: Critical Thinking and Collaboration across the Curriculum*. Charles C. Thomas Press, Springfield, Illinois (USA). 15

Aggarwal, C. C., Wolf, J. L., Yu, P. S., Procopiuc, C., and Park, J. S. (1999). Fast Algorithms for Projected Clustering. In Delis, A., Faloutsos, C., and Ghandeharizadeh, S., editors, *Proceedings of the Twenty-Fifth ACM SIGMOD International Conference on Management of Data*, pages 61–72, Philadelphia, Pennsylvania (USA). Association for Computer Machinery (ACM), ACM Press. 45

Aggarwal, C. C. and Yu, P. S. (2000). Finding Generalized Projected Clusters in High Dimensional Spaces. In Chen, W., Naughton, J. F., and Bernstein, P. A., editors, *Proceedings of the Twenty-Sixth ACM SIGMOD International Conference on Management of Data*, pages 70–781, Dallas, Texas (USA). Association for Computer Machinery (ACM), ACM Press. 45

Agrawal, R., Gehrke, J., Gunopulos, D., and Raghavan, P. (1998). Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications. In Haas, L. M. and Tiwary, A., editors, *Proceedings of the Twenty-Fourth ACM SIGMOD International Conference on Management of Data*, pages 94–105, Seattle, Washington (USA). Association for Computer Machinery (ACM), ACM Press. 45, 62

Aidos, H. and Fred, A. L. N. (2011a). Hierarchical Clustering with High Order Dissimilarities. In Perner, P., editor, *Proceedings of the Seventh International Conference on Machine learning and Data Mining in Pattern Recognition (MLDM)*, pages 280–293, New York, New York (USA). Institute of Computer Vision and applied Computer Sciences (IBaI), Springer. 66, 82, 89, 90, 91, 94, 95, 118

Aidos, H. and Fred, A. L. N. (2011b). On the Distribution of Dissimilarity Increments. In Vitria, J., Sanches, J. M., and Hernández, M., editors, *Proceedings of the Fifth Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)*, pages 192–199, Las Palmas de Gran Canaria (Spain)). International Association for Pattern Recognitio (IAPR), Springer. 89

Aidos, H. and Fred, A. L. N. (2011c). Statistical Modeling of Dissimilarity Increments for d-Dimensional Data: Application in Partitional Clustering. *Pattern Recognition*, 44(9):3061–3071. 66, 89

Akaike, H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–722. 59

Akoglu, L., Tong, H., Meeder, B., and Faloutsos, C. (2012). PICS: Parameter-free Identification of Cohesive Subgroups in Large Attributed Graphs. In *Proceedings the Twelfth SIAM International Conference on Data Mining (SDM)*, pages 439–450, Anaheim, California (USA). Society for Industrial and Applied Mathematics (SIAM), SIAM/Omnipress. 64

Al-Sultan, K. (1995). A Tabu Search Approach to the Clustering Problem. *Pattern Recognition*, 28(9):1443–1451. 45

Ally, M. (2008). Foundations of Educational Theory for Online Learning. In Anderson, T., editor, *The Theory and Practice of Online Learning*, chapter 1, pages 15–44. Athabasca University Press, Athabasca, Alberta (Canada). 2, 4, 5

Althaus, S. L. (1997). Computer-Mediated Communication in the University Classroom: An Experiment with On-Line Discussions. *Communication Education*, 46(3):158–174. 2

Amershi, S. and Conati, C. (2009). Combining Unsupervised and Supervised Classification to Build User Models for Exploratory Learning Environments. *Journal of Educational Data Mining (JEDM)*, 1(1):18–71. 24, 25

Anagnostopoulos, I., Anagnostopoulos, C., Rouskas, A., Kormentzas, G., and Vergados, D. (2006). The Wisconsin Breast Cancer Problem: Diagnosis and TTR/DFS Time Prognosis Using Probabilistic and Generalised Regression Information Classifiers. *Oncology Reports*, 15(4):975–981. 153

Andrews, L. C. (1998). *Special Functions of Mathematics for Engineers*. Oxford University Press, Oxford, England (UK), 2nd edition. 89

Antunes, C. (2011). Anticipating Students' Failure as Soon as Possible. In Romero, C., Ventura, S., Pechenizkiy, M., and Baker, R. S. J. d., editors, *Handbook of Educational Data Mining*, chapter 25, pages 353–363. Chapman & Hall/CRC Press, Taylor & Francis Group, Boca Raton, Florida (USA). 208

Aulls, M., Ibrahim, A., Pelaez, S., Wang, X., and Orjuela-Laverde, M. (2010). What Happens as Learning during Asynchronous Text-Based Discussions in an Online Learning System? In Murray, E., editor, *Proceedings of the Fifth Conference of Learning International Networks Consortium (LINC)*, pages 620–628, Cambridge, Massachusetts (USA). Massachusetts Institute of Technology (MIT), Learning International Networks Consortium (LINC). 4

Bajcsy, P. and Ahuja, N. (1998). Location- and Density-Based Hierarchical Clustering Using Similarity Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(9):1011–1015. 63

Baker, R. S. J. d. and Yacef, K. (2009). The State of Educational Data Mining in 2009: A Review and Future Visions. *Journal of Educational Data Mining (JEDM)*, 1(1):3–17. 19

Bales, R. F. and Cohen, S. P. (1979). *SYMLOG: A System for the Multiple Level Observation of Groups*. Free Press, Simon & Schuster, New York, New York (USA). 11

Ball, G. H. and Hall, D. J. (1965). ISODATA, a Novel Method of Data Analysis and Classification. Technical Report AD699616, Stanford University, Stanford, California (USA). 42, 60

Bannan-Ritland, B. (2002). Computer-Mediated Communication, eLearning, and Interactivity: A Review of the Research. *The Quarterly Review of Distance Education*, 3(2):161–179. 5, 14

Barab, S. A., Bowdish, B. E., and Lawless, K. A. (1997). Hypermedia Navigation: Profiles of Hypermedia Users. *Educational Technology Research and Development*, 45(3):23–41. 25, 26

Barab, S. A., MaKinster, J. G., Moore, J. G., Cunningham, J. A., and the ILF Design Team (2001). Designing and Building an Online Community: The Struggle to Support Sociability in the Inquiry Learning Forum. *Educational Technology Research and Development*, 49(4):71–96. 16

Barbará, D., Couto, J., and Li, Y. (2002). COOLCAT: An Entropy-Based Algorithm for Categorical Clustering. In Nicholas, C., editor, *Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM)*, pages 582–589, McLean, Virginia (USA). Association for Computer Machinery (ACM), ACM Press. 43

Bauer, J. F. (2002). Assessing Student Work from Chatrooms and Bulletin Boards. *New Directions for Teaching and Learning*, 2002(91):31–36. 3, 5

Bayá, A. E. and Granitto, P. M. (2011). Improved Gene Expression Clustering with the Parameter-Free PKNNG Metric. In de Souza, O. N., Telles, G. P., and Palakal, M. J., editors, *Proceedings of the Sixth Brazilian Conference on Advances in Bioinformatics and Computational Biology (BSB)*, pages 50–57, Brasilia (Brazil). Springer. 64

Bay, S. D. and Pazzani, M. J. (1999). Detecting Change in Categorical Data: Mining Contrast Sets. In Fayyad, U. M., Chaudhuri, S., and Madigan, D., editors, *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 302–306, San Diego, California (USA). Association for Computer Machinery (ACM), ACM Press. 43

Beaudoin, M. F. (2002). Learning or Lurking? Tracking the 'Invisible' Online Student. *The Internet and Higher Education*, 5(2):147–155. 11

Bellman, R. E. (1961). *Adaptive Control Processes: A Guided Tour*. Princeton University Press, Princeton, New Jersey (USA). 124, 177

Ben-Hur, A., Horn, D., Siegelmann, H., and Vapnik, V. (2001). Support Vector Clustering. *Journal of Machine Learning Research (JMLR)*, 2:125–137. 45, 62

Bennett, K. P. and Mangasarian, O. L. (1992). Robust Linear Programming Discrimination of Two Linearly Inseparable sets. *Optimization Methods and Software*, 1(1):23–34. 153

Bento, R., Brownstein, B., Kemery, E., and Zacur, S. R. (2005). A Taxonomy of Participation in Online Courses. *Journal of College Teaching & Learning (TLC)*, 2(12):79–86. 12, 13

Bento, R. and Schuster, C. (2003). Participation: The Online Challenge. In Aggarwal, A., editor, *Web-Based Education: Learning from Experience*, chapter 10, pages 156–164. Idea Group Publishing, Hershey, Pennsylvania (USA). 6

Berge, Z. L. (1999). Interaction in Post-Secondary Web-Based Learning. *Educational Technology*, 39(1):5–11. 15

Berge, Z. L. (2002). Active, Interactive, and Reflective eLearning. *The Quarterly Review of Distance Education*, 3(2):181–190. 16

Berkhin, P. (2006). A Survey of Clustering Data Mining Techniques. In Kogan, J., Nicholas, C., and Teboulle, M., editors, *Grouping Multidimensional Data: Recent Advances in Clustering*, chapter 2, pages 25–71. Springer, New York, New York (USA). 23, 41, 47

Beuchot, A. and Bullen, M. (2005). Interaction and Interpersonality in Online Discussion Forums. *Distance Education*, 26(1):67–87. 7, 8, 11, 12, 13

Beyer, K., Goldstein, J., Ramakrishnan, R., and Shaft, U. (1999). When Is "Nearest Neighbor" Meaningful? In Beeri, C. and Buneman, P., editors, *Proceedings of the Seventh International Conference on Database Theory (ICDT)*, pages 217–235, Jerusalem (Israel). Institute of Computer Science, The Hebrew University, Springer. 45, 125, 177

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, New York (USA). 38, 42

Böhlke, O. (2003). A Comparison of Student Participation Levels by Group Size and Language Stages during Chatroom and Face-to-Face Discussions in German. *CALICO Journal*, 21(1):67–87. 8

Böhm, C., Faloutsos, C., Pan, J. Y., and Plant, C. (2007). RIC: Parameter-Free Noise-Robust Clustering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(3):2–30. 64, 66

Böhm, C., Goebl, S., Oswald, A., Plant, C., Plavinski, M., and Wackersreuther, B. (2010). Integrative Parameter-Free Clustering of Data with Mixed Type Attributes. In Zaki, M. J., Yu, J. X., Ravindran, B., and Pudi, V., editors, *Proceedings of the Fourteenth Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD)*, volume 1, pages 38–47, Hyderabad (India). International Institute of Information Technology (IIIT), Springer. 64

Biggs, J. B. (1987). *Student Approaches to Learning and Studying*. Australian Council for Educational Research, Melbourne (Australia). 26

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, England (UK). 46

Bishop, J. (2011). Transforming Lurkers into Posters The Role of the Participation Continuum. In Grout, V., Picking, R., Oram, D., Cunningham, S., and Houlden, N., editors, *Proceedings of the Fourth International Conference on Internet Technologies and Applications (ITA)*, pages 25–32, Wrexham, North Wales (UK). Centre for Applied Internet Research, Glyndwr University. 11

Bliuc, A. M., Ellis, R., Goodyear, P., and Piggott, L. (2010). Learning through Face-to-Face and Online Discussions: Associations between Students' Conceptions, Approaches and Academic Performance in Political Science. *British Journal of Educational Technology (BJET)*, 41(3):512–524. 26

Bowes, J. (2002). Building Online Communities for Professional Networks. In Stubbs, T., editor, *Proceedings of the Global Summit of Online Knowledge Networks*, pages 71–74, Adelaide (Australia). Education.au Limited. 11

Brachman, R. and Anand, T. (1996). The Process of Knowledge Discovery in Databases: A Human-Centered Approach. In Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R., editors, *Advances in Knowledge Discovery and Data Mining*, chapter 2, pages 37Ű–58. AAAI Press, Menlo Park, California (USA). 18

Branon, R. F. and Essex, C. (2001). Synchronous and Asynchronous Communication Tools in Distance Education: A Survey of Instructors. *TechTrends*, 45(1):36–42. 3, 4

Brown, R. E. (2001). The Process of Community-Building in Distance Learning Classes. *Journal of Asynchronous Learning Networks (JALN)*, 5(2):18–35. 4, 5

Brush, A. J. B., Wang, X., Turner, T. C., and Smith, M. A. (2005). Assessing Differential Usage of Usenet Social Accounting Meta-Data. In van der Veer, G. and Gale, C., editors, *Proceedings of the Twenty-Third SIGCHI Conference on Human Factors in Computing Systems*, pages 889–898, Portland, Oregon (USA). Association for Computer Machinery (ACM), ACM Press. 14

Bullen, M. (1998). Participation and Critical Thinking in Online University Distance Education. *The Journal of Distance Education*, 13(2):1–32. 8, 9

Burkhalter, B. and Smith, M. (2004). Inhabitant's Uses and Reactions to Usenet Social Accounting Data. In Snowden, D. N., Churchill, E. F., and Frécon, E., editors, *Inhabited Information Spaces: Living with your Data*, chapter 15, pages 291–305. Springer, New York, New York (USA). 13

Burnett, C. (2003). Learning to Chat: Tutor Participation in Synchronous Online Chat. *Teaching in Higher Education*, 8(2):247–261. 3

Calvani, A., Fini, A., Molino, M., and Ranieri, M. (2010). Visualizing and Monitoring Effective Interactions in Online Collaborative Groups. *British Journal of Educational Technology (BJET)*, 41(2):213–226. 8, 9, 19, 20, 21, 22, 181, 208

Campos, M., Laferrière, T., and Harasim, L. (2001). The Post-Secondary Networked Classroom: Renewal of Teaching Practices and Social Interaction. *Journal of Asynchronous Learning Networks (JALN)*, 5(2):36–52. 5

Cao, Y. and Wu, J. (2002). Projective ART for Clustering Data Sets in High Dimensional Spaces. *Neural Networks*, 15(1):105–120. 46

Carliner, S. (1999). *An Overview of Online Learning*. Human Resource Development Press, Amherst, Massachusetts (USA). 2

Carlsson, G. and Mémoli, F. (2009). Characterization, Stability and Convergence of Hierarchical Clustering Methods. *Journal of Machine Learning Research (JMLR)*, 11:1425–1470. 54, 70

Carpenter, G. and Grossberg, S. (1987). A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine. *Computer Vision, Graphics and Image Processing*, 37(1):54–115. 46, 62

Carpenter, G., Grossberg, S., and Rosen, D. (1991). Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System. *Neural Networks*, 4(6):759Ű771. 46

Caspi, A., Chajuta, E., and Saportaa, K. (2008). Participation in Class and in Online Discussions: Gender Differences. *Computers & Education*, 50(3):718–724. 6, 7

Cesario, E., Manco, G., and Ortale, R. (2007). Top-Down Parameter-Free Clustering of High-Dimensional Categorical Data. *IEEE Transactions on Knowledge and Data Engineering*, 19(12):1607–1624. 64

Chakrabarti, D. (2004). AutoPart: Parameter-Free Graph Partitioning and Outlier Detection. In Boulicaut, J. F., Esposito, F., Giannotti, F., and Pedreschi, D., editors, *Proccedings of the Eigth European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, pages 112–124, Pisa (Italy). Istituto di Scienza e Tecnologie dell'Informazione (ISTI), Università di Pisa, Springer. 64

Chakrabarti, D., Papadimitriou, S., Modha, D. S., and Faloutsos, C. (2004). Fully Automatic Cross-Associations. In Kim, W., Kohavi, R., Gehrke, J., and DuMouchel, W., editors, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 79–88, Seattle, Washington (USA). Association for Computer Machinery (ACM), ACM Press. 64

Chan, J., Hayes, C., and Daly, E. (2010). Decomposing Discussion Forums and Boards Using User Roles. In Cohn, A., editor, *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 215–218, Washington, D.C. (USA). Association for the Advancement of Artificial Intelligence (AAAI), AAAI Press. 27, 31

Chang, J. W. and Jin, D. S. (2002). A New Cell-Based Clustering Method for Large, High-Dimensional Data in Data Mining Applications. In Cohn, A., editor, *Proceedings of the Seventeenth ACM Symposium on Applied Computing (SAC)*, pages 503–507, Madrid (Spain). Association for Computer Machinery (ACM), ACM Press. 44

Cheeseman, P. and Stutz, J. (1996). Bayesian Classification (Autoclass): Theory and Results. In Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R., editors, *Advances in Knowledge Discovery and Data Mining*, chapter 6, pages 153–180. AAAI Press, Menlo Park, California (USA). 43

Chen, J. S., Ching, R. K. H., and Lin, Y. S. (2004). An Extended Study of the K-means Algorithm for Data Clustering and its Applications. *Journal of the Operational Research Society (JORS)*, 55(9):976–987. 63

Chen, X., Liu, W., Qiu, H., and Lai, J. (2011). APSCAN: A Parameter Free Algorithm for Clustering. *Pattern Recognition Letters*, 32(7):973–986. 65, 66

Cheng, C., Fu, A., and Zhang, Y. (1999). Entropy-Based Subspace Clustering for Mining Numerical Data. In Fayyad, U. M., Chaudhuri, S., and Madigan, D., editors, *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 84–93, San Diego, California (USA). Association for Computer Machinery (ACM), ACM Press. 62

Cheung, Y. M. and Jia, H. (2013). Categorical-and-Numerical-Attribute Data Clustering Based on a Unified Similarity Metric without Knowing Cluster Number. *Pattern Recognition*, 46(8):2228–2238. 66

Chiang, J. and Hao, P. (2003). A New Kernel-Based Fuzzy Clustering approach: Support Vector Clustering with Cell Growing. *IEEE Transactions on Fuzzy Systems*, 11(4):518–527. 45

Cho, H., Stefanone, M., and Gay, G. (2002). Social Network Analysis of Information Sharing Networks in a CSCL Community. In Stahl, G., editor, *Proceedings of the Fourth International Conference on Computer-Supported Collaborative Learning (CSCL)*, pages 43–50, Boulder, Colorado (USA). International Society of the Learning Sciences (ISLS), Lawrence Erlbaum Associates. 19

Chowdhury, A. K. M. R., Mollah, M. E., and Rahman, M. A. (2010). An Efficient Method for Subjectively Choosing Parameter 'k' Automatically in VDBSCAN (Varied Density Based Spatial Clustering of Applications with Noise) Algorithm. In Mahadevan, V. and Jianhong, Z., editors, *Proceedings of the Second International Conference on Computer and Automation Engineering (ICCAE)*, volume 1, pages 38–41, Singapore (Republic of Singapore). IEEE Computer Society. 65

Chu, S. and Roddick, J. (2000). A Clustering Algorithm Using the Tabu Search Approach with Simulated Annealing. In Ebecken, N. and Brebbia, C., editors, *Data Mining II - Proceedings of the Second International Conference on Data Mining Methods and Databases*, pages 515–523, Cambridge, England (UK). 45

Cobo, G., García-Solórzano, D., Morán, J. A., Santamaría, E., Monzo, C., and Melenchón, J. (2012). Using Agglomerative Hierarchical Clustering to Model Learner Participation Profiles in Online Discussion Forums. In Shum, S. B., Gasevic, D., and Ferguson, R., editors, *Proceedings of the Second International Conference on Learning Analytics and Knowledge (LAK)*, pages 248–251, Vancouver, British Columbia (Canada). University of British Columbia, ACM Press. 177

Cobo, G., García-Solórzano, D., Santamaría, E., Morán, J. A., Melenchón, J., and Monzo, C. (2011). Modelling Students Activity in Online Discussion Forums: A Strategy Based on Time Series and Agglomerative Hierarchical Clustering. In Pechenizkiy, M., Calders, T., Conati, C., Ventura, S., Romero, C., and Stamper, J. C., editors, *Proceedings of the Fourth International Conference on Educational Data Mining (EDM)*, pages 253–258, Eindhoven (The Netherlands). International Educational Data Mining Society (IEDMS, Eindhoven University of Technology (TU/e printservice). 177

Cobo, G., Sevillano, X., Alías, F., and Socoró, J. C. (2006). Técnicas de representación de textos para clasificación no supervisada de documentos. In *Proceedings of the Twenty-Second Congress of the Spanish Society of Natural Language Processing (SEPLN)*, volume 37, pages 320–336, Zaragoza (Spain). Sociedad Española del Procesamiento del Lenguaje Natural (SEPLN). 163

Cole, R. A. (2000). *Issues in Web-Based Pedagogy: A Critical Primer*. Greenwood Press, Westport, Connecticut (USA). 2

Collins, M. (1998). The Use of Email and Electronic Bulletin Boards in College-Level Biology. *Journal of Computers in Mathematics and Science Teaching (JCMST)*, 17(1):75–94. 4

Conover, W. J. (1980). *Practical Nonparametric Statistics*. John Wiley & Sons, Hoboken, New Jersey (USA), 3rd edition. 51, 52

Cooper, L. (2000). Online Courses: Tips for Making them Work. *Technological Horizons in Education (THE Journal)*, 27(8):86–92. 4

Crawford, K., Gordon, S., Nicholas, J., and Prosser, M. ((1998a). Qualitatively Different Experiences of Learning Mathematics at the University. *Learning and Instruction*, 8(5):455–468. 26

Dash, M., Lin, K., and Xu, X. (2001). '1+1>2': Merging Distance and Density Based Clustering. In Werner, B., editor, *Proceedings of the Seventh International Conference on Database Systems for Advanced Applications*, pages 32–39, Hong Kong (China). City University of Hong Kong, IEEE Computer Society. 44

Dave, R. N. (1996). Validating Fuzzy Partitions 0btained through C-Shells Clustering. *Pattern Recognition Letters*, 17(6):613–623. 55

Dave, R. N. and Krishnapuram, R. (1997). Robust Clustering Methods: A Unified View. *IEEE Transactions of Fuzzy Systems*, 5(3):270–293. 47

Davidson-Shivers, G. V., Muilenburg, L. Y., and Tanner, E. J. (2001). How Do Students Participate in Synchronous and Asynchronous Online Discussions? *Journal of Educational Computing Research*, 25(4):341–366. 3, 7, 8

Davies, J. and Graff, M. (2005). Performance in E-Learning: Online Participation and Student Grades. *British Journal of Educational Technology (BJET)*, 36(4):657–663. 6, 7

Day, W. H. E. and Edelsbrunner, H. (1984). Efficient algorithms for Agglomerative Hierarchical Clustering Methods. *Journal of Classification*, 1(1):7–24. 74

de Haan, M. (2002). Distributed Cognition and the Shared Knowledge Model of the Mazahua: A Cultural Approach. *Journal of Interactive Learning Research (JILR)*, 13(1):31–50. 16

de Laat, M., Lally, V., Lipponen, L., and Simons, R. J. (2007). Investigating Patterns of Interaction in Networked Learning and Computer-Supported Collaborative Learning: A Role for Social Network Analysis. *International Journal of Computer-Supported Collaborative Learning (IJCSCL)*, 2(1):87–103. 8, 19, 20, 21, 22

December, J. (1996). Units of Analysis for Internet Communication. *Journal of Communication*, 46(1):14–38. 2

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by Latent Semantic Indexing. *Journal of the American Society for Information Science and Technology (JASIST)*, 41(6):391–407. 163

Defays, D. (1977). An Efficient Algorithm for a Complete Link Method. *The Computer Journal*, 20(4):364–366. 73

del Valle, R. and Duffy, T. M. (2009). Online Learning: Learner Characteristics and Their Approaches to Managing Learning. *Instructional Science*, 37(2):129–149. 25, 27, 29, 31, 51, 52

Delattre, M. and Hansen, P. (1980). Bicriterion Cluster Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(4):277–291. 71

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38. 43

Denoeud, L., Garreta, H., and Guénoche, A. (2006). Comparison of Distance Indices Between Partitions. In Batagelj, V., Bock, H. H., Ferligoj, A., and Žiberna, A., editors, *Data Science and Classification*, chapter 3, pages 21–28. Springer, New York, New York (USA). 48

Dimitriadou, E., Dolnicar, S., and Weingessel, A. (2002). An Examination of Indexes for Determining the Number of Clusters in Binary Data Sets. *Psychometrika*, 67(1):137–159. 55

Ding, C. and He, X. (2002). Cluster Merging and Splitting in Hierarchical Clustering Algorithms. In Kumar, V., Tsurnoto, S., Zhong, N., Yu, P. S., and Wu, X., editors, *Proceedings of the Second*

International Conference on Data Mining (ICDM), pages 139–146, Maebashi City (Japan). IEEE Computer Society. 63

Donath, J. S. (1996). Identity and Deception in the Virtual Community. In Kollock, P. and Smith, M. A., editors, *Communities in Cyberspace*, chapter 2, pages 29–59. Routledge, Taylor & Francis Group, London, England (UK). 13

Dringus, L. P. and Ellis, T. (2005). Using Data Mining as a Strategy for Assessing Asynchronous Discussion Forums. *Computers & Education*, 45(1):141–160. 9

Dubes, R. C. (1987). How Many Clusters Are Best? - An Experiment. *Pattern Recognition*, 20(6):645–663. 55

Dubes, R. C. (1993). Cluster Analysis and Related Issues. In Chen, C. H., Pau, L. S., and Wang, P. S. P., editors, *Handbook of Pattern Recognition and Computer Vision*, chapter 1, pages 3–32. World Scientific Publishing, Singapore (Republic of Singapore). 48, 55

Dubes, R. C. and Jain, A. K. (1979). Validity Studies in Clustering Methodologies. *Pattern Recognition*, 11(4):235–254. 48, 120

DuBien, J. and Warde, W. (1979). A Mathematical Comparison of the Members of an Infinite Family of Agglomerative Clustering Algorithms. *The Canadian Journal of Statistics*, 7(1):29–38. 71

Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern Classification*. John Wiley & Sons, Hoboken, New Jersey (USA). 35, 51, 52

Dunlosky, J. (1998). Linking Metacognitive Theory to Education. In Hacker, D. J., Dunlosky, J., and Graesser, A. C., editors, *Metacognition in Educational Theory and Practice*, chapter Epilogue, pages 367–381. Lawrence Erlbaum Associates, Mahwah, New Jersey (USA). 17

Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3(3):32–57. 38, 42

Dy, J. G. and Brodley, C. E. (2004). Feature Selection for Unsupervised Learning. *Journal of Machine Learning Research (JMLR)*, 5:845–889. 45, 56

El-Sonbaty, Y., Ismail, M. A., and Farouk, M. (2004). An Efficient Density Based Clustering Algorithm for Large Databases. In Azada, D., editor, *Proceedings of the Sixteenth IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 673–677, Boca Raton, Florida (USA). Information Technology Research Institute, Wright State University, IEEE Computer Society. 44

Ellis, A. (2003). Personality Type and Participation in Networked Learning Environments. *Educational Media International*, 40(1–2):101–114. 7, 9

Erlin, B., Yusof, N., and Rahman, A. A. (2009). Students' Interactions in Online Asynchronous Discussion Forum: A Social Network Analysis. In O'Conner, L., editor, *Proceedings of the First International Conference on Education Technology and Computer (ICETC)*, pages 25–29, Singapore (Republic of Singapore). International Association of Computer Science and Information Technology (IACSIT), IEEE Computer Society. 8, 9, 19, 21

Ertöz, L., Steinbach, M., and Kumar, V. (2002). A New Shared Nearest Neighbor Clustering Algorithm and its Applications. In Dhillon, I. and .Kogan, J., editors, *Workshop on Clustering High Dimensional Data and its Applications, in the Second SIAM International Conference on Data Mining (SDM)*, pages 105–115, Arlington, Virginia, (USA). Society for Industrial and Applied Mathematics (SIAM). 44

Ertöz, L., Steinbach, M., and Kumar, V. (2003). Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data. In Barbará, D. and Kamath, C., editors, *Proceedings of the Third SIAM International Conference on Data Mining (SDM)*, pages 47–58, San Francisco, California, (USA). Society for Industrial and Applied Mathematics (SIAM). 44

Ester, M., Kriegel, H. P., Sander, J., and Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Data Sets with Noise. In Simoudis, E., Han, J., and Fayyad, U., editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 226–231, Portland, Oregon (USA). Association for the Advancement of Artificial Intelligence (AAAI), AAAI Press. 44, 61, 65

Everitt, B. S., Landau, S., Leese, M., and Stahl, D. (2011). *Cluster Analysis*. John Wiley & Sons, Hoboken, New Jersey (USA), 5th edition. 23, 31, 38, 41, 53, 55, 58, 72, 73, 74, 120

Fay, M. P. and Proschan, M. A. (2010). WilcoxonŰMannŰWhitney or T-test? On Assumptions for Hypothesis Tests and Multiple Interpretations of Decision Rules. *Statistics Surveys*, 4:1–39. 52

Fayyad, U. M. (1996). Data Mining and Knowledge Discovery: Making Sense out of Data. *IEEE Expert: Intelligent Systems and their Applications*, 11(5):20–25. 18, 19

Fayyad, U. M., Piatetsky-Shapiro, G., and Smyth, P. (1996). From Data Mining to Knowledge Discovery: An Overview. In Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R., editors, *Advances in Knowledge Discovery and Data Mining*, chapter 1, pages 1–30. AAAI Press, Menlo Park, California (USA). 17, 18, 19

Fenty, J. (2004). Analyzing Distances. *The Stata Journal*, 4(1):1–26. 35

Fern, X. Z. and Lin, W. (2008). Cluster Ensemble Selection. In Mohammed, J. Z. and Wang, K., editors, *Proceedings of the Eighth SIAM International Conference on Data Mining (SDM)*, pages 787–797, Atlanta, Georgia, (USA). Society for Industrial and Applied Mathematics (SIAM). 46, 58

Fisher, R. A. (1936). The Use of Multiple Measurements in Taxonomic Problems. *Annual of Eugenics*, 7(2):179–188. 151

Florek, K., Lukaszewicz, J., Steinhaus, H., and Zubrzycki, S. (1951). Sur la liason et la division des points d'un ensembre fini. *Colloquium Mathematicae*, 2(3–4):282–285. 43, 72

Fodor, I. K. (2003). A Survey of Dimension Reduction Techniques. Technical Report UCRL-ID-148494, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, California (USA). 45, 56

Forgy, E. (1965). Cluster Analysis of Multivariate Data: Efficiency vs. Interpretability of Classifications. *Biometrics*, 21:768–780. 38, 42, 167

Forina, M., Leardi, R., Armanino, C., and Lanteri, S. (1990). PARVUS: An Extendable Package of Programs for Data Exploration, Classification and Correlation. *Journal of Chemometrics*, 4(2):191–193. 149

Fraley, C. and Raftery, A. (1998). How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *The Computer Journal*, 41(8):578–588. 43, 59

Fred, A. L. N. (2001). Finding Consistent Clusters in Data Partitions. In Kittler, J. and Roli, F., editors, *Proceedings of the Second International Workshop on Multiple Classifier Systems (MCS)*, pages 309–318, Cambridge, England (UK). Center for Vision, Speech and Signal Processing of the University of Surrey and Department of Electrical and Electronic Engineering of the University of Cagliari, Springer. 48, 49

Fred, A. L. N. and Jain, A. K. (2003). Robust Data Clustering. In Martin, D., editor, *Proceedings of the Thirteenth IIEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 128–133, Madison, Wisconsin (USA). IEEE Computer Society. 46

Fred, A. L. N. and Leitão, J. M. N. (2000). Clustering Under a Hypothesis of Smooth Dissimilarity Increments. In Sanfeliu, A., Villanueva, J. J., Vanrell, M., Alquezar, R., Jain, A. K., and Kittler, J., editors, *Proceedings of the Fifteenth International Conference on Pattern Recognition (ICPR)*, volume 2, pages 190–194, Barcelona (Spain). IEEE Computer Society. 82, 84, 85, 89, 94, 95, 96, 101, 114

Fred, A. L. N. and Leitão, J. M. N. (2003). A New Cluster Isolation Criterion Based on Dissimilarity Increments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):944–958. 66, 82, 83, 85, 86, 88, 91, 94, 100, 101, 118, 120, 141

Fritsch, H. (1999). Host contacted, waiting for reply. In Bernath, U. and Rubin, E., editors, *Final Report and Documentation of the Virtual Seminar for Professional Development in Distance Education*, chapter 12, pages 355–378. Institute for Distance Education, University of Maryland, College Park, Maryland (USA). 11

Fulford, C. P. and Zhang, S. (1993). Perceptions of Interaction: The Critical Predictor in Distance Education. *American Journal of Distance Education (AJDE)*, 7(3):8–21. 2

Gan, G., Ma, C., and Wu, J. (2007). *Data Clustering: Theory, Algorithms and Applications*. American Statistical Association and the Society for Industrial and Applied Mathematics (ASA-SIAM), Philadelphia, Pennsylvania (USA). 35, 37, 39, 40, 42, 43, 44, 45, 47, 53, 56, 59, 70, 71, 72, 73, 74, 75, 76, 177

Ganti, V., Gehrke, J., and Ramakrishnan, R. (1999). CACTUS: Clustering Categorical Data Using Summaries. In Fayyad, U. M., Chaudhuri, S., and Madigan, D., editors, *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 73–83, San Diego, California (USA). Association for Computer Machinery (ACM), ACM Press. 43

García-Solórzano, D., Cobo, G., Santamaría, E., Morán, J. A., Monzo, C., and Melenchón, J. (2012). Educational Monitoring Tool Based on Faceted Browsing and Data Portraits. In Shum, S. B., Gasevic, D., and Ferguson, R., editors, *Proceedings of the Second International Conference on Learning Analytics and Knowledge (LAK)*, pages 170–178, Vancouver, British Columbia (Canada). University of British Columbia, ACM Press. 208

Garrison, D. R. (1989). *Understanding Distance Education: A Framework for the Future*. Routledge, Taylor & Francis Group, London, England (UK). 6

Garrison, D. R. and Anderson, T. (2003). *E-Learning in the 21st Century: A Framework for Research and Practice*. Routledge, Taylor & Francis Group, London, England (UK). 2

Garrison, D. R., Anderson, T., and Archer, W. (2000). Critical Inquiry in a Text-Based Environment: Computer Conferencing in Higher Education. *The Internet and Higher Education*, 2(2–3):87–105. 8, 11

Garton, L., Haythornthwaite, C., and Wellman, B. (1997). Studying Online Social networks. *Journal of Computer-Mediated Communication (JCMC)*, 3(1):529–530. 19

Gath, I. and Geva, A. B. (1989a). Fuzzy Clustering for the Estimation of the Parameters of the Components of Mixtures of Normal Distributions. *Pattern Recognition Letters*, 9(2):77–86. 43

Gath, I. and Geva, A. B. (1989b). Unsupervised Optimal Fuzzy Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):773–781. 43, 55

Geva, A. B. (1999). Hierarchical Unsupervised Fuzzy Clustering. *IEEE Transactions of Fuzzy Systems*, 7(6):723–733. 40, 42, 47, 60, 82

Gibson, D., Kleinberg, J., and Raghavan, P. (2000). Clustering Categorical Data: An Approach Based on Dynamical Systems. *The VLDB Journal*, 8(3-4):222–236. 43

Gionis, A., Mannila, H., and Tsaparas, P. (2007). Clustering Aggregation. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):1–30. 58

Girolami, M. (2002). Mercer Kernel-Based Clustering in Feature Space. *IEEE Transactions on Neural Networks*, 13(3):780–784. 56

Gitman, I. (1972). A Parameter-Free Clustering Model. *Pattern Recognition*, 4(3):307–315. 63

Golder, S. (2003). *A Typology of Social Roles in Usenet*. PhD thesis, Department of Linguistics, Harvard University, Cambridge, Massachusetts (USA)). 14

Gordon, A. D. (1987). A Review of Hierarchical Classification. *Journal of the Royal Statistical Society. Series A (General)*, 150(2):119–137. 70

Gordon, A. D. (1996). Hierarchical Classification. In Arabie, P., Hubert, L. J., and de Soete, G., editors, *Clustering and Classification*, chapter 2, pages 65–121. World Scientific, River Edge, New Jersey (USA). 53, 70

Gordon, A. D. (1998). Cluster Validation. In Hayashi, C., Ohsumi, N., Yajima, K., Tanaka, Y., Bock, H., and Bada, Y., editors, *Data Science, Classification, and Related Methods*, chapter 2, pages 22–39. Springer, New York, New York (USA). 48

Gower, J. C. (1967). A Comparison of Some Methods of Cluster Analysis. *Biometrics*, 23(4):623–637. 42, 75

Grabusts, P. and Borisov, A. (2002). Using Grid-Clustering Methods in Data Classification. In Kawada, S., editor, *Proceedings of the Second International Conference on Parallel Computing in Electrical Engineering (PARELEC)*, pages 425–428, Warsaw (Poland). Polish-Japanese Institute of Information Technology, IEEE Computer Society. 44

Graham, M. and Scarborough, H. (1999). Computer Mediated Communication and Collaborative Learning in an Undergraduate Distance Education Environment. *Australian Journal of Educational Technology (AJET)*, 15(1):20–46. 14

Gray, R. (2002). Assessing Students' Written Projects. *New Directions for Teaching and Learning*, 2002(91):37–43. 5

Grünwald, P., Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. (1998). Minimum Encoding Approaches for Predictive Modeling. In Cooper, G. F. and Moral, S., editors, *Proceedings of the Fourteenth International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 183–192, Madison, Wisconsin (USA). University of Wisconsin Business School, Morgan Kaufmann Publishers. 59, 64

Guha, S., Rastogi, R., and Shim, K. (1998). CURE: An Efficient Clustering Algorithm for Large Databases. In Haas, L. M. and Tiwary, A., editors, *Proceedings of the Twenty-Fourth ACM SIGMOD International Conference on Management of Data*, pages 73–84, Seattle, Washington (USA). Association for Computer Machinery (ACM), ACM Press. 42

Guha, S., Rastogi, R., and Shim, K. (2000). ROCK: A Robust Clustering Algorithm for Categorical Attributes. *Information Systems*, 25(5):345–366. 43

Gunawardena, C. N. and McIsaac, M. S. (2004). Distance Education. In Jonassen, D. H., editor, *Handbook of Research for Educational Communications and Technology*, chapter 14, pages 355–395. Lawrence Erlbaum Associates, Mahwah, New Jersey (USA), 2nd edition. 2

Hakkarainen, K. and Palonen, T. (2003). Patterns of Female and Male Students' Participation in Peer Interaction in Computer-Supported Learning. *Computers & Education*, 40(4):327–342. 8

Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2001). On Clustering Validation Techniques. *Journal of Intelligent Information Systems (JIIS)*, 17(2–3):107–145. 47, 48

Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2002a). Cluster Validity Methods: Part I. *ACM SIGMOD Record*, 31(2):40–45. 18, 46, 48, 50

Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2002b). Cluster Validity Methods: Part II. *ACM SIGMOD Record*, 31(3):19–27. 47, 48, 50

Hall, L., Özyurt, I., and Bezdek, J. (1999). Clustering with a Genetically Optimized Approach. *IEEE Transactions on Evolutionary Computation*, 3(2):103–112. 45

Hammah, R. and Curran, J. (2000). Validity Measures for the Fuzzy Cluster Analysis of Orientations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1467–1472. 47

Hammami, N. and Bedda, M. (2010). Improved Tree Model for Arabic Speech Recognition. In *Proceedings of the Third International Conference on Computer Science and Information Technology (ICCSIT)*, volume 5, pages 521–526, Chengdu (China). IEEE Computer Society. 158

Hammond, M. (1999). Issues Associated with Participation in Online Forums – The Case of the Communicative Learner. *Education and Information Technologies*, 4(4):353–367. 10

Hara, N., Bonk, C. J., and Angeli, C. (2000). Content Analysis of Online Discussion in an Applied Educational Psychology Course. *Instructional Science*, 28(2):115–152. 22

Hara, N. and Kling, R. (2000). Students' Distress with a Web-Based Distance Education Course: An Ethnographic Study of Participants' Experiences. *Information, Communication & Society*, 3(4):557–579. 14

Harasim, L. (1989). On-Line Education: A New Domain. In Mason, R. and Kaye, A. R., editors, *Mindweave: Communication, Computers and Distance Education*, chapter 4, pages 50–62. Pergamon Press, Oxford, England (UK). 2

Harris, N. and Sandor, M. (2007). Developing Online Discussion Forums as Student Centred Peer E-Learning Environments. In Atkinson, R. J., McBeath, C., Soong, S. K. A., and Cheers, C., editors, *Proceedings of the Twenty-Fourth Annual Conference of the Australian Society for Computers in Learning in Tertiary Education (ASCILITE)*, pages 383–387, Singapore (Republic of Singapore). Centre for Educational Development, Nanyang Technological University, Australian Society for Computers in Learning in Tertiary Education (ASCILITE). 4

He, J., Tong, H., Papadimitriou, S., Eliassi-Rad, T., Faloutsos, C., and Carbonell, J. (2009). PaCK: Scalable Parameter-Free Clustering on K-Partite Graphs. In Davidson, I. and Domeniconi, C., editors, *Seventh Workshop on Link Analysis, Counter-terrorism and Security (LACS) in the Ninth SIAM International Conference on Data Mining (SDM)*, Sparks, Nevada, (USA). Society for Industrial and Applied Mathematics (SIAM). 64

Heller, K. A. and Ghahramani, Z. (2005). Bayesian Hierarchical Clustering. In De Raedt, L. and Wrobel, S., editors, *Proceedings of the Twenty-Second International Conference on Machine Learning (ICML)*, pages 297–304, Bonn (Germany). The International Machine Learning Society (IMLS), ACM Press. 43

Henri, F. (1992). Computer Conferencing and Content Analysis. In Kaye, A. R., editor, *Collaborative Learning through Computer conferencing*, chapter 8, pages 117–136. Springer, New York, New York (USA). 11

Herbin, M., Bonnet, N., and Vautrot, P. (2001). Estimation of the Number of Clusters and Influence Zones. *Pattern Recognition Letters*, 22(14):1557–1568. 56

Herring, S. C. (2004). Slouching Toward the Ordinary: Current Trends in Computer-Mediated Communication. *New Media & Society*, 6(1):26–36. 13

Higgins, R. N. (1991). *Computer-Mediated Cooperative Learning: Synchronous and Asynchronous Communication between Students Learning Nursing Diagnosis*. PhD thesis, University of Toronto, Toronto, Ontario (Canada). 11

Hillman, D. C., Willis, D. J., and Gunawardena, C. N. (1994). Learner Interface Interaction in Distance Education: An Extension of Contemporary Models and Strategies for Practitioners. *American Journal of Distance Education (AJDE)*, 8(2):30–42. 2

Himelboim, I., Gleave, E., and Smith, M. (2009). Discussion Catalysts in Online Political Discussions: Content Importers and Conversation Starters. *Journal of Computer-Mediated Communication (JCMC)*, 14(4):771–789. 22

Hines, R. A. and Pearl, C. E. (2004). Increasing Interaction in Web-Based Instruction: Using Synchronous Chats and Asynchronous Discussions. *Rural Special Education Quarterly*, 23(2):33–36. 3

Hinneburg, A. and Keim, D. (1998). An Efficient Approach to Clustering in Large Multimedia Data Sets with Noise. In Agrawal, R. and Stolorz, P., editors, *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 58–65, New York, New York (USA). Association for the Advancement of Artificial Intelligence (AAAI), AAAI Press. 44, 61

Hinneburg, A. and Keim, D. (1999). Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering. In Atkinson, M. P., Orlowska, M. E., Valduriez, P., Zdonik, S. B., and Brodie, M. L., editors, *Proceedings of the 25th International Conference on Very*

Large Data Bases (VLDB), pages 506–517, Edinburgh, Scotland (UK). Edinburgh Napier University, Morgan Kaufmann Publishers. 44, 61

Hirai, H., Chou, B. H., and Suzuki, E. (2011). A Parameter-Free Method for Discovering Generalized Clusters in a Network. In Elomaa, T., Hollmén, J., and Mannila, H., editors, *Proceedings of the Fourteenth International Conference on Discovery Science (DS)*, pages 135–149, Espoo (Finland). Aalto University, Springer. 64

Hofmann, T. and Buhmann, J. (1997). Pairwise Data Clustering by Deterministic Annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1):1–14. 45

Holman, E. (1992). Statistical Properties of Large Published Classifications. *Journal of Classification*, 9(2):187–210. 70

Hrastinski, S. (2006). The Relationship between Adopting a Synchronous Medium and Participation in Online Group Work: An Explorative Study. *Interactive Learning Environments*, 14(2):137–152. 9

Hrastinski, S. (2008). What is Online Learner Participation? A Literature Review. *Computers & Education*, 51(4):1755–1765. 6, 181

Huang, H., Mok, P. Y., Kwok, Y. L., and Au, S. C. (2009). A Parameter Free Approach for Clustering Analysis. In Jiang, X. and Petkov, N., editors, *Proceedings of the Thirteenth International Conference on Computer Analysis of Images and Patterns (CAIP)*, pages 157–164, Münster (Germany). University of Münster, Springer. 63

Hubert, L. J. (1974). Some Applications of Graph Theory to Clustering. *Psychometrika*, 39(3):283–309. 42

Hyvärinen, A., Karhunen, J., and Oja, E. (2001). *Independent Component Analysis*. John Wiley & Sons, Hoboken, New Jersey (USA). 45

Ienco, D., Pensa, R. G., and Meo, R. (2009). Parameter-Free Hierarchical Co-clustering by n-Ary Splits. In Buntine, W. L., Grobelnik, M., Mladenic, D., and Shawe-Taylor, J., editors, *Proccedings of the Thirteenth European Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD)*, volume 1, pages 580–595, Bled (Slovenia). Jožef Stefan Institut (JSI), Springer. 64

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Data*. Prentice Hall, Upper Saddle River, New Jersey (USA). 23, 37, 38, 39

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data Clustering: A Survey. *ACM Computing Surveys*, 31(3):264–323. 23, 35, 37, 38, 41, 43, 46, 53, 54

Jambu, M. (1978). *Classification automatique pour l'analyse de données*. Éditions Dunod, Paris (France). 71

Joachims, T. (1996). A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. Computer Science Technical Report CMU-CS-96-118, Carnegie Mellon University, Pittsburgh, Pennsylvania (USA). 162

Johnson, G. M. (2006). Synchronous and Asynchronous Text-Based CMC in Educational Contexts: A Review of Recent Research. *TechTrends*, 50(4):46–53. 3

Johnson, H. (2007). Dialogue and the Construction of Knowledge in E-Learning: Exploring Students' Perceptions of Their Learning While Using Blackboard's Asynchronous Discussion Board. *European Journal of Open, Distance and E-Learning (EURODL)*, 10(1):13–20. 5

Johnson, S. C. (1967). Hierarchical Clustering Schemes. *Psychometrika*, 32(3):241–254. 43, 72

Jolliffe, I. T. (1986). *Principal Component Analysis*. Springer, New York, New York (USA). 45

Jonassen, D. H., Davidson, M., Collins, M., Campbell, J., and Haag, B. (1995). Constructivism and Computer-Mediated Communication in Distance Education. *American Journal of Distance Education (AJDE)*, 9(2):7–26. 14, 15

Jonassen, D. H. and Kwon, H. I. (2001). Communication Patterns in Computer Mediated versus Face-to-Face Problem Solving. *Educational Technology Research and Development*, 49(1):35–51. 16

Jones, M. G. and Harmon, S. W. (2002). What Professors Need to Know About Technology to Assess Online Student Learning. *New Directions for Teaching and Learning*, 2002(91):19–30. 5

Kannan, R., Vempala, S., and Vetta, A. (2004). On Clusterings: Good, Bad and Spectral. *Journal of the ACM (JACM)*, 51(3):497–515. 43

Kantardzic, M. (2003). *Data Mining: Concepts, Models, Methods and Algorithms*. IEEE Computer Society, New York, New York (USA). 17, 19

Karayan, S. and Crowe, J. (1997). Student Perceptions of Electronic Discussion Groups. *Technological Horizons in Education (THE Journal)*, 24(9):69–71. 4

Karayiannis, N., Bezdek, J., Pal, N., Hathaway, R., and Pai, P. (1996). Repairs to GLVQ: A New Family of Competitive Learning Schemes. *IEEE Transactions on Neural Networks*, 7(35):1062–1071. 46

Karypis, G., Han, E. H., and Kumar, V. (1999). Chameleon: A Hierarchical Clustering Algorithm using Dynamic Modeling. *IEEE Computer*, 32(8):68–75. 43, 61

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, Hoboken, New Jersey (USA). 38, 42

Keegan, D. (1990). *The Foundations of Distance Education*. Routledge, Taylor & Francis Group, London, England (UK). 2

Kelly, A. E. (2004). Design Research in Education: Yes, but is It Methodological? *Journal of the Learning Sciences*, 13(1):115–128. 9

Kendall, M. G. (1938). A New Measure of Rank Correlation. *Biometrika*, 30(1–2):81–89. 161

Keogh, E., Lonardi, S., and Ratanamahatana, C. A. (2004). Towards Parameter-Free Data Mining. In Kim, W., Kohavi, R., Gehrke, J., and DuMouchel, W., editors, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 206–215, Seattle, Washington (USA). Association for Computer Machinery (ACM), ACM Press. 63

Khan, B. H. (1997). Web-Based Instruction: What Is It and Why Is It? In Khan, B. H., editor, *Web-Based Instruction*, chapter 1, pages 5–18. Educational Technology Publications, Englewood Cliffs, New Jersey (USA). 2

Khan, T. M., Clear, F., and Sajadi, S. S. (2012). The Relationship between Educational Performance and Online Access Routines: Analysis of Students' Access to an Online Discussion Forum. In Shum, S. B., Gasevic, D., and Ferguson, R., editors, *Proceedings of the Second International Conference on Learning Analytics and Knowledge (LAK)*, pages 226–229, Vancouver, British Columbia (Canada). University of British Columbia, ACM Press. 28, 51, 52

Kim, A. J. (2000). *Community Building on the Web: Secret Strategies for Successful Online Communities*. Peachpit Press, Pearson Education, San Francisco, California (USA). 14

Kim, J., Shaw, E., and Ravi, S. (2011). Mining Student Discussions for Profiling Participation and Scaffolding Learning. In Romero, C., Ventura, S., Pechenizkiy, M., and Baker, R. S. J. d., editors, *Handbook of Educational Data Mining*, chapter 21, pages 299–310. Chapman & Hall/CRC Press, Taylor & Francis Group, Boca Raton, Florida (USA). 5, 7, 181

Kleinberg, J. (2002). An Impossibility Theorem for Clustering. In Becker, S., Thrun, S., and Obermayer, K., editors, *Proceedings of the Sixteenth Conference on Advances in Neural Information Processing Systems (NIPS)*, volume 15, pages 463–470, Vancouver, British Columbia (Canada). The Neural Information Processing Systems Foundation, MIT Press. 54

Klemm, W. R. (1997). Benefits of Collaboration Software for On-Site Classes. In Ho, C. P., editor, *Proceedings of the Second Annual Teaching in Community Colleges (TCC) On-Line Conference*, Honolulu, Hawaii (USA). Kapi'olani Community College, University of Hawaii, ERIC Clearinghouse. 4, 5

Knowlton, D. S. (2000). A Theoretical Framework for the Online Classroom: A Defense and Delineation of a Student-Centered Pedagogy. *New Directions for Teaching and Learning*, 2000(84):5–14. 14

Knowlton, D. S. (2003a). Evaluating College Students' Efforts in Asynchronous Discussion: A Systematic Process. *The Quarterly Review of Distance Education*, 4(1):31–41. 16, 17

Knowlton, D. S. (2003b). Preparing Students for Educated Living: Virtues of Problem-Based Learning across the Higher Education Curriculum. In Knowlton, D. S. and Sharp, D. C., editors, *Problem-Based Learning in the Information Age*, chapter 1, pages 5–12. Jossey-Bass Publishers, San Francisco, California (USA). 16, 17

Knowlton, D. S. (2005). A Taxonomy of Learning through Asynchronous Discussion. *Journal of Interactive Learning Research (JILR)*, 16(2):155–177. 5, 14

Knuth, D. E. (1998). *The Art of Computer Programming*, volume 3: Sorting and Searching. Addison-Wesley, Pearson Education, Boston, Massachusetts (USA), 2nd edition. 88

Kohonen, T. (1990). The Self-Organizing Map. *Proceedings of the IEEE*, 78(9):1464–1480. 46

Kollock, P. and Smith, M. (1996). Managing the Virtual Commons: Cooperation and Conflict in Computer Communities. In Herring, S. C., editor, *Computer Mediated Communication: Linguistic, Social, and Cross-Cultural Perspectives*, chapter 6, pages 109–128. John Benjamins Publishing Company, Amsterdam (The Netherlands). 11

Kothari, R. and Pitts, D. (1999). On Finding the Number of Clusters. *Pattern Recognition Letters*, 20(4):405–416. 56

Kotsiantis, S. and Pintelas, P. (2004). Recent Advances in Clustering: A Brief Survey. *WSEAS Transactions on Information Science and Applications*, 1(1):73–81. 40, 41

Kruskal, W. H. and Wallis, W. A. (1952). Use of Ranks in One-Criterion Variance Analysis. *Journal of the American Statistical Association (JASA)*, 47(260):583–621. 51, 52

Kuboni, O. and Martin, A. (2004). An Assessment of Support Strategies used to Facilitate Distance Students' Participation in a Web-Based Learning Environment in the University of the West Indies. *Distance Education*, 25(1):7–29. 7, 9

Laghos, A. and Zaphiris, P. (2006). Sociology of Student-Centred e-Learning Communities: A Network Analysis. In Isaías, P., McPherson, M., and Bannister, F., editors, *Proceedings of the Fourth IADIS International Conference on e-Society*, Trinity College, Dublin (Ireland). International Association for Development of the Information Society (IADIS). 19

Lance, G. N. and Williams, W. T. (1966). A Generalized Sorting Strategy for Computer Classifications. *Nature*, 212(5058):218. 43, 73

Lance, G. N. and Williams, W. T. (1967). A General Theory of Classificatory Sorting Strategies: 1. Hierarchical Systems. *The Computer Journal*, 9(1):373–380. 43, 71, 72

Land, S. M. and Dornisch, M. M. (2002). A Case Study of Student Use of Asynchronous Bulletin Board Systems (BBS) To Support Reflection and Evaluation. *Journal of Educational Technology Systems*, 30(4):365–377. 4

Landsberger, J. (2001). Integrating a Web-Based Bulletin Board into your Class: A Guide for Faculty. *TechTrends*, 45(5):50–53. 4, 5

Langfelder, P., Zhang, B., and Horvath, S. (2008). Defining Clusters from a Hierarchical Cluster Tree: The Dynamic Tree Cut Package for R. *Bioinformatics*, 20(5):719–720. 56, 80, 82

Larson, B. E. (2000). Classroom Discussion: A Method of Instruction and a Curriculum Outcome. *Teaching and Teacher Education*, 16(5-6):661–677. 4

Laurillard, D. (1999). A Conversational Framework for Individual Learning Applied to the 'Learning Organisation' and the 'Learning Society'. *Systems Research and Behavioural Science*, 16(2):113–122. 2

Laurillard, D. (2002). *Rethinking University Teaching: A Conversational Framework for the Effective Use of Learning Technologies*. Routledge, Taylor & Francis Group, London, England (UK). 4, 5

Lave, J. and Wenger, E. (1991). *Situated Learning: Legitimate Peripheral participation*. Cambridge University Press, Cambridge, England (UK). 14

Lee, J. and McKendree, J. (1999). Learning Vicariously in a Distributed Environment. *Journal of Active Learning*, 10(1):4–10. 11

Leinonen, P. and Järvelä, S. (2003). Individual Students' Interpretations of their Contribution to the Computer-Mediated Discussions. *Journal of Interactive Learning Research (JILR)*, 14(1):99–122. 16

Lerman, I. C. (1981). *Classification et analyse ordinale des données*. Éditions Dunod, Paris (France). 80

Lewis, D. C., Treves, J. A., and Shindlin, A. B. (1997). Making Sense of Academic Cyberspace: Case Study of an Electronic Classroom. *College Teaching*, 45(3):96–100. 16

Lin, X. (2001). Designing metacognitive activities. *Educational Technology Research and Development*, 49(2):23–40. 16

Lipponen, L., Rahikainen, M., Hakkarainen, K., and Palonen, T. (2002). Effective Participation and Discourse through a Computer Network: Investigating Elementary Students' Computer Supported Interaction. *Journal of Educational Computing Research*, 27(4):355–384. 8

Lipponen, L., Rahikainen, M., Lallimo, J., and Hakkarainen, K. (2003). Patterns of Participation and Discourse in Elementary Students' Computer-Supported Collaborative Learning. *Learning and Instruction*, 13(5):487–509. 7, 19, 20

Liu, M. and Bera, S. (2005). An Analysis of Cognitive Tool Use Patterns in a Hypermedia Learning Environment. *Educational Technology Research and Development*, 53(1):5–21. 25

Liu, P., Zhou, D., and Wu, N. (2007). VDBSCAN: Varied Density Based Spatial Clustering of Applications with Noise. In *Proceedings of the Fourth International Conference on Service Systems and Service Management (ICSSSM)*, volume 1, pages 528–531, Chengdu (China). IEEE Computer Society. 61, 65

López, M. I., Luna, J. M., Romero, C., and Ventura, S. (2012). Classification Via Clustering for Predicting Final Marks Based on Student Participation in Forums. In Yacef, K., Zaïane, O., Hershkovitz, H., Yudelson, M., and Stamper, J. C., editors, *Proceedings of the Fifth International Conference on Educational Data Mining (EDM)*, pages 148–151, Chania (Greece). International Educational Data Mining Society (IEDMS). 25

Lundgren, D. C. (1977). Developmental Trends in the Emergence of Interpersonal Issues in T Groups. *Small Group Research (SGR)*, 8(2):179–200. 11

Mabry, E. A. (1997). Framing Flames: The Structure of Argumentative Messages on the Net. *Journal of Computer-Mediated Communication (JCMC)*, 2(4). 11

MacKnight, C. B. (2000). Teaching Critical Thinking through Online Discussions. *Educause Quarterly*, 23(4):38–41. 4, 5

MacQueen, J. (1967). Some Methods for Classification and Analysis of Multivariate Observations. In Le Cam, L. M. and Neyman, J., editors, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, Berkeley, California (USA). Statistical Laboratory of the University of California, Berkeley, University of California Press. 38, 42, 167

Maimon, O. and Rokach, L. (2005). *Data Mining and Knowledge Discovery Handbook*. Springer, New York, New York (USA). 18

Mangasarian, O. L., Street, W. N., and Wolberg, W. H. (1995). Breast Cancer Diagnosis and Prognosis Via Linear Programming. *Operations Research*, 43(4):570–577. 155

Marcoccia, M. (2004). On-line Polylogues: Conversation Structure and Participation Framework in Internet Newsgroups. *Journal of Pragmatics*, 36(1):115–145. 13

Martínez, W. L., Martínez, A. R., and Solka, J. L. (2005). *Exploratory Data Analysis with MATLAB*. Chapman & Hall/CRC Press, Taylor & Francis Group, Boca Raton, Florida (USA). 43

Mason, R. (1994). *Using Communication Media in Open and Flexible Learning*. Kogan Page, London, England (UK). 9, 10, 11

Masters, K. and Oberprieler, G. (2004). Encouraging Equitable Online Participation through Curriculum Articulation. *Computers & Education*, 42(4):319–332. 8

Mazzolini, M. and Maddison, S. (2003). Sage, Guide or Ghost? The Effect of Instructor Intervention on Student Participation in Online Discussion Forums. *Computers & Education*, 40(3):237–253. 7

McCampbell, B. (2000). Toys or Tools? On-Line Bulletin Boards and Chat Rooms. *Principal Leadership (High School Edition)*, 1(3):73–74. 4

McLachlan, G. and Krishnan, T. (1997). *The EM Algorithm and Extensions*. John Wiley & Sons, Hoboken, New Jersey (USA). 43, 59

McLachlan, G. and Peel, D. (2000). *Finite Mixture Models*. John Wiley & Sons, Hoboken, New Jersey (USA). 59

McLinden, M., McCall, S., Hinton, D., and Weston, A. (2006). Participation in Online Problem Based Learning: Insights from Postgraduate Teachers Studying through Open and Distance Education. *Distance Education*, 27(3):331–353. 9

McLoughlin, C. (2002). Computer Supported Teamwork: An Integrative Approach to Evaluating Cooperative Learning in an Online Environment. *Australian Journal of Educational Technology (AJET)*, 18(2):227–254. 3

McLoughlin, C. and Luca, J. (2001). Quality in Online Delivery: What does It Mean for Assessment in E-learning environment? In Kennedy, G., Keppell, M., McNaught, C., and Petrovic, T., editors, *Proceedings of the Eighteenth Annual Conference of the Australian Society for Computers in Learning in Tertiary Education (ASCILITE)*, pages 417–426, Melbourne (Australia). The University of Melbourne, Australian Society for Computers in Learning in Tertiary Education (ASCILITE). 4

McLoughlin, C. and Panko, M. (2002). Multiple Perspectives on the Evaluation of Online Discussion. In Barker, P. and Rebelsky, S., editors, *Proceedings of the Fourteenth World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA)*, pages 24–29, Denver, Colorado (USA). Association for the Advancement of Computing in Education (AACE), EdITLib Digital Library. 5

McQuitty, L. L. (1957). Elementary Linkage Analysis for Isolating Orthogonal and Oblique Types and Typal Relevancies. *Educational and Psychological Measurement (EPM)*, 17(2):207–222. 43, 72

McQuitty, L. L. (1966). Similarity Analysis by Reciprocal Pairs for Discrete and Continuous Data. *Educational and Psychological Measurement (EPM)*, 27(1):21–46. 73

Meng, X. and van Dyk, D. (1997). The EM Algorithm – An Old Folk-Song Sung to a Fast New Tune. *Journal of the Royal Statistical Society. Series B (Methodological)*, 59(3):511–567. 43

Milligan, G. W. (1996). Clustering Validation Results and Implications for Applied Analyses. In Arabie, P., Hubert, L. J., and de Soete, G., editors, *Clustering and Classification*, chapter 10, pages 341–375. World Scientific, River Edge, New Jersey (USA). 55

Milligan, G. W. and Cooper, M. C. (1985). An Examination of Procedures for Determining the Number of Clusters in a Data Set. *Psychometrika*, 50(2):159–179. 55, 56, 58

Molina, L. C., Belanche, L., and Nebot, . (2002). Feature Selection Algorithms: A Survey and Experimental Evaluation. In Kumar, V., Tsurnoto, S., Zhong, N., Yu, P. S., and Wu, X., editors, *Proceedings of the Second International Conference on Data Mining (ICDM)*, pages 306–313, Maebashi City (Japan). IEEE Computer Society. 45, 56

Moore, M. G. (1989). Three Types of Interaction. *American Journal of Distance Education (AJDE)*, 3(2):1–6. 2

Morris, M. and Ogan, C. (1996). The Internet as Mass Medium. *Journal of Communication*, 46(1):39–50. 11

Morrison, G. R. and Guenther, P. F. (2000). Designing Instruction for Learning in Electronic Classrooms. *New Directions for Teaching and Learning*, 2000(84):15–22. 5

Mu, J., Fei, H., and Dong, X. (2008). A Parameter-Free Clustering Algorithm Based on Density Model. In *Proceedings of the Ninth International Conference for Young Computer Scientists (ICYCS)*, pages 1825–1831, Zhang Jia Jie, Hunan (China). IEEE Computer Society. 65

Mueller, N. S., Haegler, K., Shao, J., Plant, C., and Böhm, C. (2011). Weighted Graph Compression for Parameter-free Clustering With PaCCo. In *Proceedings the Eleventh SIAM International Conference on Data Mining (SDM)*, pages 932–943, Mesa, Arizona, (USA). Society for Industrial and Applied Mathematics (SIAM), SIAM/Omnipress. 64

Muirhead, B. (2002). Salmon's E-tivities: The key to Active Online Learning. *The United States Distance Learning Association (USDLA) Journal*, 16(8):53–57. 5

Murtagh, F. (1983). A Survey of Recent Advances in Hierarchical Clustering Algorithms. *The Computer Journal*, 26(4):354–359. 72, 73, 74

Murtagh, F. (1984). Complexities of Hierarchic Clustering Algorithms: State of the Art. *Computational Statistics Quarterly (CSQ)*, 1:101–113. 74

Murtagh, F. (1985). *Multidimensional Clustering Algorithms*. Physica-Verlag, Würzburg (Germany). 72, 73, 74

Murtagh, F. (2007). The Haar Wavelet Transform of a Dendrogram. *Journal of Classification*, 24(1):3–32. 80

Murtagh, F. and Contreras, P. (2012). Algorithms for Hierarchical Clustering: An Overview. *Wiley Interdisciplinary Reviews (WIREs): Data Mining and Knowledge Discovery*, 2(1):86–97. 41, 44, 70, 71, 72, 73, 75, 76, 79, 82

Nagpal, P. B. and Mann, P. A. (2011). Comparative Study of Density-Based Clustering Algorithms. *International Journal of Computer Applications (IJCA)*, 27(11):44–47. 65

Nandi, D., Chang, S., and Balbo, S. (2009). A Conceptual Framework for Assessing Interaction Quality in Online Discussion Forums. In Atkinson, R. J. and McBeath, C., editors, *Proceedings*

of the Twenty-Sixth Annual Conference of the Australian Society for Computers in Learning in Tertiary Education (ASCILITE), pages 665–673, Auckland (New Zealand). Auckland University of Technology (AUT), The University of Auckland, Australian Society for Computers in Learning in Tertiary Education (ASCILITE). 9

Ngomo, A. C. N. (2010). Parameter-Free Clustering of Protein-Protein Interaction Graphs. In Džeroski, S., Rogers, S., and Sanguinetti, G., editors, *Proceedings of the Fourth International Workshop on Machine Learning in Systems Biology (MLSB)*, pages 43–46, Edinburgh, Scotland (UK). University of Edinburgh, Pattern Analysis, Statistical Modelling and Computational Learning (PASCAL). 64

Ngwenya, J., Annand, D., and Wang, E. (2008). Supporting Asynchronous Discussions among Online Learners. In Anderson, T. and Elloumi, F., editors, *The Theory and Practice of Online Learning*, chapter 13, pages 319–347. Athabasca University Press, Athabasca, Alberta (Canada). 8, 208

Nicholson, S. A. and Bond, N. (2003). Collaborative Reflection and Professional Community Building: An Analysis of Preservice Teachers' Use of an Electronic Discussion Board. *Journal of Technology and Teacher Education (JTATE)*, 11(2):259–279. 15

Nonnecke, B. and Preece, J. (1999). Shedding Light on Lurkers in Online Communities. In Buckner, K., editor, *Proceedings of Esprit Intelligent Information Interfaces (i3) Workshop on Ethnographic Studies in Real and Virtual Environments: Inhabited Information Spaces and Connected Communities*, pages 123–128, Edinburgh, Scotland (UK). LiMe & eSCAPE, Queen Margaret University College. 11

Nonnecke, B. and Preece, J. (2000). Lurker Demographics: Counting the silent. In Turner, T., Szwillus, G., Czerwinski, M., Peterno, F., and Pemberton, S., editors, *Proceedings of the Eighteenth SIGCHI Conference on Human Factors in Computing Systems*, pages 73–80, The Hague (The Netherlands). Association for Computer Machinery (ACM), ACM Press. 11

Nonnecke, B. and Preece, J. (2001). Why Lurkers Lurk. In Gorgone, J. and Fedorowicz, J., editors, *Proceedings of the Seventh Americas Conference on Information Systems (AMCIS)*, Boston, Massachusetts (USA). Association for Information Systems (AIS), AIS Electronic Library (AISeL). 11

Nonnecke, B., Preece, J., Andrews, D., and Voutour, R. (2004). Online Lurkers Tell Why. In Gorgone, J. and Fedorowicz, J., editors, *Proceedings of the Tenth Americas Conference on Information Systems (AMCIS)*, New York, New York (USA). Association for Information Systems (AIS), AIS Electronic Library (AISeL). 11

Oh, H. K., Yoon, S. H., and Kim, S. W. (2012). Hierarchical Clustering and Outlier Detection for Effective Image Data Organization. In Lee, S. H., Hanzo, L., Ismail, R., Kim, D. S., Chung, M. Y., and Lee, S. W., editors, *Proceeding of the Sixth International Conference on Ubiquitous Infor-*

mation Management and Communication (ICUIMC), Kuala Lumpur (Malaysia). Association for Computer Machinery (ACM), ACM Press. 64

Oliver, J., Baxter, R., and Wallace, C. (1996). Unsupervised Learning Using MML. In Saitta, L., editor, *Proceedings of the Thirteenth International Conference on Machine Learning (ICML)*, pages 364–372, Bari (Italy). The International Machine Learning Society (IMLS), Morgan Kaufmann Publishers. 59

Pal, N. and Bezdek, J. (1995). On Cluster Validity for the Fuzzy-means Model. *IEEE Transactions on Fuzzy Systems*, 3(3):370–379. 47

Palloff, R. M. and Pratt, K. (1999). *Building Learning Communities in Cyberspace: Effective Strategies for the Online Classroom*. Jossey-Bass Publishers, San Francisco, California (USA). 15, 17

Papadimitriou, S., Sun, J., Faloutsos, C., and Yu, P. S. (2008). Hierarchical, Parameter-Free Community Discovery. In Daelemans, W., Goethals, B., and Morik, K., editors, *Proccedings of the Twelfth European Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD)*, volume 2, pages 170–187, Antwerp (Belgium). Universiteit Antwerpen (UA), Springer. 64

Parimala, M., Lopez, D., and Senthilkumar, N. C. (2011). A Survey on Density Based Clustering Algorithms for Mining Large Spatial Databases. *International Journal of Advanced Science and Technology (IJAST)*, 31:59–66. 65

Park, N. H. and Lee, W. S. (2004). Statistical Grid-Based Clustering over Data Streams. *ACM SIGMOD Record*, 33(1):32–37. 44

Paulus, T. (2007). CMC Modes for Learning Tasks at a Distance. *Journal of Computer-Mediated Communication (JCMC)*, 12(4):1322–1345. 3

Pelleg, D. and Moore, A. (2000). X-means: Extending K-means with Efficient Estimation of the Number of Clusters. In Langley, P., editor, *Proceedings of the Seventeenth International Conference on Machine Learning (ICML)*, pages 727–734, Stanford, California (USA). The International Machine Learning Society (IMLS), Morgan Kaufmann Publishers. 42, 59

Preece, J., Nonnecke, B., and Andrews, D. (2003). The Top Five Reasons for Lurking: Improving Community Experiences for Everyone. *Computers in Human Behavior*, 20(2):201–223. 11

Prestera, G. E. and Moller, L. A. (2001). Exploiting Opportunities for Knowledge-Building in Asynchronous Distance Learning Environments. *The Quarterly Review of Distance Education*, 2(2):93–104. 5, 15, 16

Rabbany, R., Takaffoli, M., and Zaïane, O. R. (2011). Analyzing Participation of Students in Online Courses Using Social Network Analysis Techniques. In Pechenizkiy, M., Calders, T., Conati, C.,

Ventura, S., Romero, C., and Stamper, J. C., editors, *Proceedings of the Fourth International Conference on Educational Data Mining (EDM)*, pages 21–30, Eindhoven (The Netherlands). International Educational Data Mining Society (IEDMS, Eindhoven University of Technology (TU/e printservice). 19, 20, 21, 208

Rafaeli, S., Ravid, G., and Soroka, V. (2004). De-lurking in Virtual Communities: A Social Communication Network Approach to Measuring the Effects of Social and Cultural Capital. In Sprague, R. H., editor, *Proceedings of the Thirty-Seventh Hawaii International Conference on System Sciences (HICSS)*, Big Island, Hawaii (USA). University of Hawaii at Manoa, IEEE Computer Society. 11

Raju, B. H. V. S. R. and Kumari, V. V. (2011). Comparison of Parameter Free MST Clustering Algorithm with Hierarchical Agglomerative Clustering Algorithms. *International Journal of Computer Applications (IJCA)*, 34(4):26–31. 64

Rapoport, A. and Fillenbaum, S. (1972). An Experimental Study of Semantic Structures. In Romney, A. K., Shepard, R. N., and Nerlove, S. B., editors, *Multidimensional Scaling; Theory and Applications in the Behavioral Sciences: Applications*, volume 2, pages 93–131. Seminar Press, New York, New York (USA). 80

Ravi, S. and Kim, J. (2007). Profiling Student Interactions in Threaded Discussions with Speech Act Classifiers. In Luckin, R., Koedinger, K. R., and Greer, J. E., editors, *Proceedings of the Thirteenth Artificial Intelligence in Education Conference (AIED)*, pages 357–364, Los Angeles, California (USA). The International Artificial Intelligence in Education Conference (AIED) Society, IOS Press. 8

Reffay, C. and Chanier, T. (2003). How Social Network Analysis Can Help to Measure Cohesion in Collaborative Distance-Learning. In Wasson, B., Ludvigsen, S., and Hoppe, U., editors, *Proceedings of the Fifth International Conference on Computer-Supported Collaborative Learning (CSCL)*, pages 343–352, Bergen (Norway). International Society of the Learning Sciences (ISLS), Kluwer Academic Publishers. 19

Relan, A. and Gillani, B. J. (1997). Web-Based Instruction and the Traditional Classroom: Similarities and Differences. In Khan, B. H., editor, *Web-Based Instruction*, chapter 4, pages 25–37. Educational Technology Publications, Englewood Cliffs, New Jersey (USA). 2

Rendón, E., Abundez, I., Arizmendi, A., and Quiroz, E. M. (2011). Internal versus External Cluster Validation Indexes. *International Journal of Computers and Communications*, 5(1):27–34. 48

Reuven, A., Zippy, E., Gilad, R., and Aviva, G. (2003). Network Analysis of Knowledge Construction in Asynchronous Learning Networks. *Journal of Asynchronous Learning Networks (JALN)*, 7(3):1–23. 22

Rezaee, M. R., Lelieveldt, B. P. F., and Reibe, J. H. C. (1998). A New Cluster Validity Index for the Fuzzy C-mean. *Pattern Recognition Letters*, 19(3–4):237–246. 55

Rissanen, J. (1996). Fisher Information and Stochastic Complexity. *IEEE Transactions on Information Theory*, 42(1):40–47. 59, 64

Romero, C., López, M. I., Luna, J. M., and Ventura, S. (2013). Predicting Students' Final Performance from Participation in On-Line Discussion Forums. *Computers & Education*, 68:458–472. 181, 208

Romero, C. and Ventura, S. (2007). Educational Data Mining: A Survey from 1995 to 2005. *Expert Systems with Applications*, 33(1):135–146. 19

Romero, C., Ventura, S., Pechenizkiy, M., and Baker, R. S. J. d. (2011). Introduction. In Romero, C., Ventura, S., Pechenizkiy, M., and Baker, R. S. J. d., editors, *Handbook of Educational Data Mining*, chapter 1, pages 1–6. Chapman & Hall/CRC Press, Taylor & Francis Group, Boca Raton, Florida (USA). 19

Romiszowski, A. and Mason, R. (2004). Computer-Mediated Communication. In Jonassen, D. H., editor, *Handbook of Research for Educational Communications and Technology*, chapter 15, pages 397–432. Lawrence Erlbaum Associates, Mahwah, New Jersey (USA), 2nd edition. 3

Rosenberg, M. J. (2001). *E-Learning: Strategies for Delivering Knowledge in the Digital Age*. McGraw-Hill, New York, New York (USA). 2

Ross, J. A. (1996). The Influence of Computer Communication Skills on Participation in a Computer Conferencing Course. *Journal of Educational Computing Research*, 15(1):37–52. 6

Rossett, A. (2002). Waking in the Night and Thinking about E-Learning. In Rossett, A., editor, *The ASTD E-Learning Handbook*, chapter 1, pages 3–18. McGraw-Hill, New York, New York (USA). 2

Rossman, M. H. (1999). Successful Online Teaching Using an Asynchronous Learner Discussion Forum. *Journal of Asynchronous Learning Networks (JALN)*, 3(2):91–97. 4

Rourke, L. and Anderson, T. (2002). Using Peer Teams to Lead Online Discussions. *Journal of Interactive Media in Education (JIME)*, 7(2):1–21. 4

Rourke, L., T. Anderson, D. R. G., and Archer, W. (1999). Assessing Social Presence in Asynchronous, Text-Based Computer Conferencing. *The Journal of Distance Education*, 14(2):50–71. 11

Rousseeuw, P. J. (1987). Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, 20(1):53–65. 49, 50

Rovai, A. P. (2000). Online and Traditional Assessments: What is the Difference? *The Internet and Higher Education*, 3(3):141–151. 5

Rovai, A. P. (2001). Building Classroom Community at a Distance: A Case Study. *Educational Technology Research and Development*, 49(4):33–48. 15, 16

Sabau, A. S. (2012). Variable Density Based Genetic Clustering. In Negru, V., editor, *Proceedings of the Fourteenth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 200–206, Timisoara (Romania). IEEE Computer Society. 65, 66

Sackville, A. and Sherratt, C. (2006). Styles of Discussion: Online Facilitation Factors. In Banks, S., Hodgson, V., Jones, C., Kemp, B., McConnell, D., and Smith, C., editors, *Proceedings of the Fifth International Conference on Networked Learning (NLC)*, Lancaster, England (UK). The University of Lancaster, The Higher Education Academy. 8

Salmon, G. (2000). *E-Moderating: The Key to Teaching and Learning Online*. Kogan Page, London, England (UK). 5, 11

Salton, G. and Buckley, C. (1988). Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, 24(5):513–523. 36, 162

Scardamalia, M. and Bereiter, C. (1996). Computer Support for Knowledge-Building Communities. In Koschmann, T. D., editor, *CSCL: Theory and Practice of an Emerging Paradigm*, chapter 10, pages 249–268. Lawrence Erlbaum Associates, Mahwah, New Jersey (USA). 4

Scheibler, D. and Schneider, W. (1985). Monte Carlo Test of the Accuracy of Cluster Analysis Algorithms - A Comparison of Hierarchical and Nonhierarchical Methods. *Multivariate Behavioral Research*, 20(3):283–304. 71

Scheibler, D. and Schneider, W. (1989). A Study of the Beta-Flexible Clustering Method. *Multivariate Behavioral Research*, 24(2):163–176. 71

Schikuta, E. (1996). Grid-Clustering: An Efficient Hierarchical Clustering Method for Very Large Data Sets. In Kavanaugh, M. E. and Werner, B., editors, *Proceedings of the Thirteenth International Conference on Pattern Recognition (ICPR)*, volume 2, pages 101–105, Vienna (Austria). IEEE Computer Society. 44

Schutz, W. C. (1994). *The Human Element: Productivity, Self-Esteem, and the Bottom Line*. Jossey-Bass Publishers, San Francisco, California (USA). 11

Schwarz, G. (1978). Estimating the dimension of a Model. *Annals of Statistics*, 6(2):461–464. 59

Scott, G., Clark, D., and Pham, T. (2001). A Genetic Clustering Algorithm Guided by a Descent Algorithm. In Kim, J. H., editor, *Proceedings of the Third Congress on Evolutionary Computation*, volume 2, pages 734–740, Seoul (South Korea). IEEE Computer Society. 45

Scott, J. P. (1991). *Social Network Analysis: A Handbook*. SAGE Publications, Thousand Oaks, California (USA). 20

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys (CSUR)*, 34(1):1–47. 162

Selim, S. and Al-Sultan, K. (1991). A Simulated Annealing Algorithm for the Clustering Problem. *Pattern Recognition*, 24(10):1003–1008. 45

Sevillano, X. (2009). *Hierarchical Self-Refining Consensus Architectures and Soft Consensus Functions for Robust Multimedia Clustering*. PhD thesis, Enginyeria i Arquitectura La Salle, Ramon Llull University, Barcelona (Spain). 17, 18, 23, 24, 34, 36, 37, 38, 42, 43, 44, 45, 46, 54, 55, 58, 59

Sevillano, X., Cobo, G., Alías, F., and Socoró, J. C. (2006a). Feature Diversity in Cluster Ensembles for Robust Document Clustering. In Efthimiadis, E. N., Dumais, S. T., Hawking, D., and Järvelin, K., editors, *Proceedings of the Twenty-Sixth International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 697–698, Seattle, Washington (USA). Association for Computer Machinery (ACM), ACM Press. 163

Sevillano, X., Cobo, G., Alías, F., and Socoró, J. C. (2006b). Robust Document Clustering by Exploiting Feature Diversity in Cluster Ensembles. In *Proceedings of the Twenty-Second Congress of the Spanish Society of Natural Language Processing (SEPLN)*, volume 37, pages 169–178, Zaragoza (Spain). Sociedad Española del Procesamiento del Lenguaje Natural (SEPLN). 163

Sevillano, X., Cobo, G., Alías, F., and Socoró, J. C. (2007a). A Hierarchical Consensus Architecture for Robust Document Clustering. In Amati, G., Carpineto, C., and Romano, G., editors, *Proceedings of the Twenty-Ninth European Conference on IR Research (ECIR)*, volume 4425, pages 741–744, Rome (Italy). The Chartered Institute for IT, Information Retrieval Special Interest Group (BCS-IRSG), Springer. 163

Sevillano, X., Cobo, G., Alías, F., and Socoró, J. C. (2007b). Text Clustering on Latent Thematic Spaces: Variants, Strengths and Weaknesses. In Davies, M. E., James, C. C., Abdallah, S. A., and Plumbley, M. D., editors, *Proceedings of the Seventh International Conference on Independent Component Analysis and Signal Separation (ICA)*, volume 4666, pages 794–801, London (UK). ICA Research Network, Springer. 163

Sharan, R. and Shamir, R. (2000). CLICK: A Clustering Algorithm with Applications to Gene Expression Analysis. In Bourne, P. E., Gribskov, M., Altman, R. B., Jensen, N., Hope, D. A., Lengauer, T., Mitchell, J. C., Scheeff, E. D., Smith, C., Strande, S., and Weissig, H., editors, *Proceedings of the Eighth International AAAI Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 307–316, La Jolla / San Diego,, California (USA). Association for the Advancement of Artificial Intelligence (AAAI), AAAI Press. 43

Sheikholeslami, G., Chatterjee, S., and Zhang, A. (1998). WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases. In Gupta, A., Shmueli, O., and Widom, J., editors, *Proceedings of the 24th International Conference on Very Large Data Bases (VLDB)*, pages 428–439, New York, New York (USA). AT&T Labs, Morgan Kaufmann Publishers. 44

Sheikholeslami, G., Chatterjee, S., and Zhang, A. (2000). WaveCluster: A Wavelet-Based Clustering Approach for Spatial Data in Very Large Databases. *The VLDB Journal*, 8(3-4):289–304. 61

Sibson, R. (1973). SLINK: An Optimally Efficient Algorithm for the Single Link Cluster Method. *The Computer Journal*, 16(1):30–34. 72

Siegel, S. and Castellan Jr., N. J. (1988). *Nonparametric Statistics for the Behavioral Sciences*. McGrawŰHill, New York, New York (USA), 2nd edition. 52

Smith, D. and Hardaker, G. (2000a). E-Learning Innovation Through the Implementation of an Internet Supported Learning Environment. *Journal of Educational Technology and Society*, 3(3):422–432. 4

Smith, D. and Hardaker, G. (2000b). Model Selection for Probabilistic Clustering Using Sross Validated Likelihood. *Statistics and Computing*, 10(1):63–72. 59

Smyth, P. (1996). Clustering Using Monte Carlo Cross-Validation. In Simoudis, E., Han, J., and Fayyad, U., editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 126–133, Portland, Oregon (USA). Association for the Advancement of Artificial Intelligence (AAAI), AAAI Press. 167

Sneath, P. H. A. (1957). The Applications of Computers to Taxonomy. *Journal of General Microbiology*, 17(1):201–226. 43, 72

Sneath, P. H. A. and Sokal, R. R. (1973). *Numerical Taxonomy: The Principles and Practice of Numerical Classification*. W. H. Freeman & Co., San Francisco, California (USA). 40, 70

Sokal, R. R. and Michener, C. D. (1958). A Statistical Method for Evaluating Systematic Relationships. *The University of Kansas Science Bulletin*, 38(2):1409–1438. 42, 43, 73, 74

Sokal, R. R. and Rohlf, F. J. (1962). The Comparison of Dendrograms by Objective Methods. *Taxon*, 11(2):33–40. 53

Sørensen, T. J. (1948). A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and its Application to Analyses of the Vegetation on Danish Commons. *Biologiske Skrifter*, 5:1–34. 72

Sousa, F. and Tendeiro, J. (2005). A Validation Methodology in Hierarchical Clustering. In Janssen, J. and Lenca, P., editors, *Proceedings of the Eleventh International Symposium on Applied Stochastic Models and Data Analysis (ASMDA)*, pages 396–403, Brest (France). École Nationale Supérieure des Télécommunications (ENST) de Bretagne. 53

Speck, B. W. (1998). The Teacher's Role in the Pluralistic Classroom. *Perspectives*, 28(1):19–43. 14

Späth, H. (1980). *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*. Ellis Horwood Publishers, West Sussex, England (UK). 38

Spurrier, J. D. (2003). On the Null Distribution of the Kruskal-Wallis Statistic. *Journal of Nonparametric Statistics*, 15(6):685–691. 52

Srinivasan, S. H. (2002). Features for Unsupervised Document Classification. In Roth, D. and van den Bosch, A., editors, *Proceedings of the Sixth Workshop on Computational Natural Language Learning (CoNLL)*, pages 36–42, Taipei (Taiwan). Association for Computational Linguistics (ACL), Morgan Kaufmann Publishers. 162, 164

Strehl, A. (2002). *Relationship-Based Clustering and Cluster Ensembles for High-Dimensional Data Mining*. PhD thesis, Faculty of the Graduate School of The University of Texas, Austin, Texas (USA). 46, 47

Strehl, A. and Ghosh, J. (2002). Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research (JMLR)*, 3:583–617. 58

Sudweeks, F. and Simoff, S. J. (1999). Complementary Explorative Data Analysis: The Reconciliation of Quantitative and Qualitative Principles. In Jones, S., editor, *Doing Internet Research: Critical Issues and Methods for Examining the Net*, chapter 2, pages 29–55. SAGE Publications, Thousand Oaks, California (USA). 11

Sun, J., Papadimitriou, S., Yu, P. S., and Faloutsos, C. (2007). GraphScope: Parameter-free Mining of Large Time-Evolving Graphs. In Berkhin, P., Caruana, R., and Wu, X., editors, *Proceedings of the Thirteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 687–696, San Jose, California (USA). Association for Computer Machinery (ACM), ACM Press. 64

Sun, T., Chen, W., Liu, Z., Wang, Y., Sun, X., Zhang, M., and Lin, C. (2011). Participation Maximization Based on Social Influence in Online Discussion Forums. In Cohn, A., editor, *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 361–368, Barcelona (Spain). Association for the Advancement of Artificial Intelligence (AAAI), AAAI Press. 3

Sundararajan, B. (2010). Emergence of the Most Knowledgeable Other (MKO): Social Network Analysis of Chat and Bulletin Board Conversations in a CSCL System. *Electronic Journal of e-Learning (EJEL)*, 8(2):191–208. 19

Takahashi, M., Fujimoto, M., and Yamasaki, N. (2007). Active Lurking: Enhancing the Value of In-house Online Communities Through the Related Practices Around the Online Communities. Working Paper 2007-006, MIT Center for Collective Intelligence (CCI), Cambridge, Massachusetts (USA). 8

Talavera, L. and Gaudioso, E. (2004). Mining Student Data to Characterize Similar Behavior Groups in Unstructured Collaboration Spaces. In de Mántaras, R. L. and Saitta, L., editors, *Proceedings of the Workshop on Artificial Intelligence in CSCL of the Sixteenth European Conference on Artificial intelligence (ECAI)*, pages 17–23, Valencia (Spain). European Coordinating Committee for Artificial Intelligence (ECCAI), IOS Press. 25

Tam, M. (2000). Constructivism, Instructional Design, and Technology: Implications for Transforming Distance Learning. *Journal of Educational Technology and Society*, 3(2):50–60. 14, 15

Tan, P. N., Steinbach, M., and Kumar, V. (2006). *Introduction to Data Mining*. Addison-Wesley, Pearson Education, Boston, Massachusetts (USA). 17, 19, 23, 31, 34, 36, 37, 38, 40, 41, 42, 47, 49, 50, 56, 57, 70

Tavangarian, D., Leypold, M., Nölting, K., Röser, M., and Voigt, D. (2004). Is E-Learning the Solution for Individual Learning? *Electronic Journal of e-Learning (EJEL)*, 2(2):273–280. 2

Taylor, J. C. (2002). Teaching and Learning Online: The Workers, The Lurkers and The Shirkers. In Jegede, O., editor, *Keynote in the Second Conference on Research in Distance & Adult Learning in Asia (CRIDALA)*, pages 1–14, Hong Kong (China). Center for Research in Distance & Adult Learning, The Open University of Hong Kong. 10, 11

Thomas, M. J. W. (2002). Learning Within Incoherent Structures: The Space of Online Discussion Forums. *Journal of Computer Assisted Learning (JCAL)*, 18(4):351–366. 5, 9

Tibshirani, R. and Knight, K. (1999). The Covariance Inflation Criterion for Adaptive Model Selection. *Journal of the Royal Statistical Society. Series B (Methodological)*, 61(3):529–546. 59

Tiene, D. (2000). Online Discussions: A Survey of Advantages and Disadvantages Compared to Face-to-Face Discussions. *Journal of Educational Multimedia and Hypermedia (JEMH)*, 9(4):369–382. 5

Tseng, L. and Yang, S. (2001). A Genetic Approach to the Automatic Clustering Problem. *Pattern Recognition*, 34(2):415–424. 45, 61

Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley, Pearson Education, Boston, Massachusetts (USA). 55

Turner, T. C. and Fisher, K. E. (2006). The Impact of Social Types within Information Communities: Findings from Technical Newsgroups. In Sprague, R. H., editor, *Proceedings of the Thirty-Ninth Hawaii International Conference on System Sciences (HICSS)*, volume 6, page 135b, Kauai, Hawaii (USA). University of Hawaii at Manoa, IEEE Computer Society. 14

Turner, T. C., Smith, M. A., Fisher, D., and Welser, H. T. (2005). Picturing Usenet: Mapping Computer-Mediated Collective Action. *Journal of Computer-Mediated Communication (JCMC)*, 10(4). 21, 22

Vandev, D. L. and Tsvetanova, Y. G. (1995). Ordered Dendrogram. In *Proceedings of the Ninth European Meeting of the Psychometric Society*, Leiden (The Netherlands). Leiden University, Faculty of social sciences, Department of data theory, Psychometric Society. 39

Vellido, A., Castro, F., and Nebot, A. (2011). Clustering Educational Data. In Romero, C., Ventura, S., Pechenizkiy, M., and Baker, R. S. J. d., editors, *Handbook of Educational Data Mining*, chapter 6,

pages 75–92. Chapman & Hall/CRC Press, Taylor & Francis Group, Boca Raton, Florida (USA). 23, 25

Viégas, F. B. and Smith, M. A. (2004). Newsgroup Crowds and AuthorLines: Visualizing the Activity of Individuals in Conversational Cyberspaces. In Sprague, R. H., editor, *Proceedings of the Thirty-Seventh Hawaii International Conference on System Sciences (HICSS)*, Big Island, Hawaii (USA). University of Hawaii at Manoa, IEEE Computer Society. 21

von Luxburg, U. (2007). A Tutorial on Spectral Clustering. Technical Report TR-149, Department for Empirical Inference, Max Planck Institute for Biological Cybernetics, Tübingen (Germany). 43

Vonderwell, S. and Zachariah, S. (2005). Factors that Influence Participation in Online Learning. *Journal of Research on Technology in Education (JRTE)*, 38(2):213–230. 7, 9

Vrasidas, C. (1999). *Meanings of Online and Face-to-Face Interactions in a Graduate Course*. PhD thesis, Arizona State University, Phoenix, Arizona (USA). 2

Vrasidas, C. and McIsaac, M. S. (1999). Factors Influencing Interaction in an Online Course. *American Journal of Distance Education (AJDE)*, 13(3):22–36. 2

Vygotsky, L. S. (1978). *Mind in Society*. Harvard University Press, Cambridge, Massachusetts (USA). 2

Wagner, E. D. (1994). In Support of a Functional Definition of Interaction. *American Journal of Distance Education (AJDE)*, 8(2):6–29. 2

Wallace, C. S. and Dowe, D. L. (1994). Intrinsic Classification by MML – The Snob Program. In *Proceedings of the Seventh Australian Joint Conference on Artificial Intelligence*, pages 37–44, Armidale (Australia). World Scientific. 43

Walther, J. B. (1996). Computer-Mediated Communication: Impersonal, Interpersonal, and Hyperpersonal Interaction. *Communication Research*, 23(1):3–43. 11, 16

Walther, J. B. and Burgoon, J. K. (1992). Relational Communication in Computer-Mediated Interaction. *Human Communication Research*, 19(1):50–88. 11

Wang, L. and Wang, Z. (2003). CUBN: A Clustering Algorithm Based on Density and Distance. In Adams, J. and Lee, J. W. T., editors, *Proceedings of the Second International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 108–112, Xi'an (China). The Northwestern Polytechnical University, IEEE Computer Society. 44

Wang, W., Yang, J., and Muntz, R. (1997). STING: A Statistical Information Grid Approach to Spatial Data Mining. In Jarke, M., Carey, M. J., Dittrich, K. R., Lochovsky, F. H., Loucopoulos, P., and Jeusfeld, M. A., editors, *Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB)*, pages 186–195, Athens (Greece). National Technical University of Athens (NTUA), Morgan Kaufmann Publishers. 44

Wang, W., Zhou, S., Ren, B., and He, S. (2013). Improved VDBSCAN with Global Optimum K. In *Proceedings of the Third International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, pages 225–228, Ostrava (Czech Republic). VSB-Technica University of Ostrava, SDIWC Digital Library. 65

Ward Jr., J. H. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association (JASA)*, 58(301):236–244. 42, 75

Ward Jr., J. H. and Hook, M. (1963). Application of a Hierarchical Grouping Procedure to a Problem of Grouping Profiles. *Educational and Psychological Measurement (EPM)*, 23(1):69–81. 42, 75

Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge, England (UK). 20

Waters, J. and Gasson, S. (2005). Strategies employed by participants in virtual communities. In Sprague, R. H., editor, *Proceedings of the Thirty-Eighth Hawaii International Conference on System Sciences (HICSS)*, page 3b, Big Island, Hawaii (USA). University of Hawaii at Manoa, IEEE Computer Society. 14

Weedman, J. (1999). Conversation and Community: The Potential of Electronic Conferences for Creating Intellectual Proximity in Distributed Learning Environments. *Journal of the American Society for Information Science and Technology (JASIST)*, 50(10):907–928. 14

Weiss, R. E. (2000). Humanizing the Online classroom. In Weiss, R. E., Knowlton, D. S., and Speck, B. W., editors, *Principles of Effective Teaching in the Online Classroom*, chapter 7, pages 47–52. Jossey-Bass Publishers, San Francisco, California (USA). 16

Welser, H. T., Gleave, E., Fisher, D., and Smith, M. (2007). Visualizing the Signatures of Social Roles in Online Discussion Groups. *Journal of Social Structure (JoSS)*, 8. 19, 20, 21, 22

Wenger, E. (1998). *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, Cambridge, England (UK). 6

Whittaker, S., Terveen, L., Hill, W., and Cherny, L. (1998). The Dynamics of Mass Interaction. In Poltrock, S. and Grudin, J., editors, *Proceedings of the Seventh Conference on Computer Supported Cooperative Work (CSCW)*, pages 257–264, Seattle, Washington (USA). Association for Computer Machinery (ACM), ACM Press. 11

Willging, P. A. (2005). Using Social Network Analysis Techniques to Examine Online Interactions. *US-China Education Review*, 2(9):46–56. 19

Williams, S. and Pury, C. (2002). Student Attitudes toward and Participation in Electronic Discussion. *International Journal of Educational Technology (IJET)*, 3(1). 8

Windham, M. and Culter, A. (1992). Information Ratios for Validating Mixture Analysis. *Journal of the American Statistical Association (JASA)*, 87(420):1188–1192. 59

Winiecki, D. J. (2003). Instructional Discussion in Online Education: Practical and Research-Oriented Perspectives. In Moore, M. G. and Anderson, W. G., editors, *Handbook of Distance Education*, chapter 14, pages 193–215. Lawrence Erlbaum Associates, Mahwah, New Jersey (USA). 4, 5

Wise, A. F., Marbouti, F., Hsiao, Y. T., and Hausknecht, S. (2012). A Survey of Factors Contributing to Learners' "Listening" Behaviors in Asynchronous Online Discussions. *Journal of Educational Computing Research*, 47(4):461–480. 29

Wise, A. F., Marbouti, F., Speer, F., and Hsiao, Y. T. (2011a). Towards an Understanding of "Listening" in Online Discussions: A Cluster Analysis of Learners' Interaction Patterns. In Spada, H., Stahl, G., Miyake, N., and Law, N., editors, *Proceedings of the Ninth International Conference on Computer-Supported Collaborative Learning (CSCL)*, volume 1, pages 88–95, Hong Kong (China). Centre for Information Technology in Education, The University of Hong Kong, International Society of the Learning Sciences (ISLS). 29

Wise, A. F., Speer, F., Marbouti, F., and Hsiao, Y. T. (2013). Broadening the Notion of Participation in Online Discussions: Examining Patterns in Learners' Online Listening Behaviors. *Instructional Science*, 41(2):323–343. 29, 31, 51, 52

Wise, A. F., Speer, J., Hsiao, Y. T., and Marbouti, F. (2011b). Factors Contributing to Learners' Online Listening Behaviors in Online and Blended Courses. In Spada, H., Stahl, G., Miyake, N., and Law, N., editors, *Proceedings of the Ninth International Conference on Computer-Supported Collaborative Learning (CSCL)*, volume 2, pages 711–715, Hong Kong (China). Centre for Information Technology in Education, The University of Hong Kong, International Society of the Learning Sciences (ISLS). 29

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, Burlington, Massachusetts (USA). 23

Wittrock, M. C. and Alesandrini, K. (1990). Generation of Summaries and Analogies and Analytic and Holistic Abilities. *American Educational Research Journal (AERJ)*, 27(3):489–502. 15

Woo, K. G., Lee, J. H., Kim, M. H., and Lee, Y. J. (2004). FINDIT: A Fast and Intelligent Subspace Clustering Algorithm Using Dimension Voting. *Information and Software Technology*, 46(4):255–271. 62

Woods, R. and Keeler, J. (2001). The Effect of Instructor's Use of Audio E-mail Messages on Student Participation in and Perceptions of Online Learning: A Preliminary Case Study. *Open Learning: The Journal of Open, Distance and e-Learning*, 16(3):263–278. 8

Wozniak, H. and Silveira, S. (2004). Online Discussions: Promoting Effective Student to Student Interaction. In Atkinson, R. J., McBeath, C., Jonas-Dwyer, D., and Phillips, R., editors, *Proceedings of the Twenty-First Annual Conference of the Australian Society for Computers in Learning*

in Tertiary Education (ASCILITE), pages 956–960, Perth (Australia). The University of Western Australia, Australian Society for Computers in Learning in Tertiary Education (ASCILITE). 5

Wu, D. and Hiltz, S. R. (2004). Predicting Learning From Asynchronous Online Discussions. *Journal of Asynchronous Learning Networks (JALN)*, 8(2):139–152. 4

Wunsch II, D., Caudell, T., Capps, C., Marks, R., and Falk, R. (1993). An Optoelectronic Implementation of the Adaptive Resonance Neural Network. *IEEE Transactions on Neural Networks*, 4(4):673–684. 46

Xiong, T., Wang, S., Mayers, A., and Monga, E. (2012). DHCC: Divisive Hierarchical Clustering of Categorical Data. *Data Mining and Knowledge Discovery*, 24(1):103–135. 66

Xu, R. and Wunsch II, D. (2005). Survey of Clustering Algorithms. *IEEE Transactions on Neural Networks*, 16(3):645–678. 23, 31, 35, 36, 38, 41, 45, 54, 55, 56, 58, 59, 60, 70, 76

Xu, R. and Wunsch II, D. (2008). *Clustering*. IEEE Computer Society, New York, New York (USA). 44

Xu, X., Ester, M., Kriegel, H. P., and Sander, J. (1998). A Distribution-Based Clustering Algorithm for Mining in Large Spatial Databases. In Urban, S. D. and Bertino, E., editors, *Proceedings of the Fourteenth International Conference on Data Engineering (ICDE)*, pages 324–331, Orlando, Florida (USA). IEEE Computer Society. 44, 65

Yang, Y. (2004). Practices of Using Discussion Board as an Assessment Tool in Online Learning Environment. In Nall, J. and Robson, R., editors, *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education (E-LEARN)*, pages 1577–1582, Washington, D.C. (USA). Association for the Advancement of Computing in Education (AACE), EdITLib Digital Library. 4, 5, 208

Yukselturk, E. (2010). An Investigation of Factors Affecting Student Participation Level in an Online Discussion Forum. *The Turkish Online Journal of Educational Technology (TOJET)*, 9(2):24–32. 3

Zahn, C. T. (1971). Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. *IEEE Transactions of Computers (TC)*, 20(1):68–86. 43, 79

Zakrzewska, D. (2008). Using Clustering Technique for Students' Grouping in Intelligent e-Learning Systems. In Holzinger, A., editor, *Proceedings of the Fourth Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society (USAB 2008)*, pages 403–410, Graz (Austria). Austrian Computer Society (OCG), Springer. 25

Zha, H., He, X., Ding, C., Simon, H., and Gu, M. (2001). Bipartite Graph Partitioning and Data Clustering. In *Proceedings of the Tenth ACM International Conference on Information and Knowledge Management (CIKM)*, pages 25–32, Atlanta, Georgia (USA). Association for Computer Machinery (ACM), ACM Press. 161

Zhang, T., Ramakrishnan, R., and Livny, M. (1996). BIRCH: An Efficient Data Clustering Method for Very Large Databases. In Jagadish, H. V. and Mumick, I. S., editors, *Proceedings of the Twenty-Second ACM SIGMOD International Conference on Management of Data*, pages 103–114, Montreal (Canada). Association for Computer Machinery (ACM), ACM Press. 42

Zhang, Y., Fu, A. W., Cai, C. H., and Heng, P. (2000). Clustering Categorical Data. In Lomet, D. B. and Weikum, G., editors, *Proceedings of the Sixteenth International Conference on Data Engineering (ICDE)*, pages 305–324, San Diego, California (USA). IEEE Computer Society. 43

Zhang, Y. and Liu, Z. (2002). Self-Splitting Competitive Learning: A New On-Line Clustering Paradigm. *IEEE Transactions on Neural Networks*, 13(2):369–380. 62

Zhao, Y., Karypis, G., and Fayyad, U. M. (2005). Hierarchical Clustering Algorithms for Document Datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168. 42

Zhao, Y. and Song, J. (2001). GDILC: A Grid-Based Density-Isoline Clustering Algorithm. In Zhong, Y. X., Shi, J., and Lin, X., editors, *Proceedings of International Conferences on Info-Tech and Info-Net (ICII)*, volume 3, pages 140–145, Beijing (China). IEEE Computer Society. 44

"Life moves pretty fast. If you don't stop and look around once in a while, you could miss it."

Ferris Bueller