Generative Manifold Learning for the Exploration of Partially Labeled Data

Raúl Cruz Barbosa

advisor
Alfredo Vellido

Departament de Llenguatges i Sistemes Informàtics

Universitat Politècnica de Catalunya

A thesis submitted for the degree of

Ph.D in Artificial Intelligence

I would like to dedicate this thesis to my loving parents, Graciela and José, and my family, Meylí, Meylí Guadalupe†, Raúl Addí and Diego.

Acknowledgements

Firstly, I would like to thank my advisor. It was a nice pleasure working with Alfredo. He gave me freedom and support throughout the development of this thesis; he shared his experience and insights with my ideas and doubts. It could not be possible my collaboration with the SOCO group at UPC trough the AIDTumour (TIN2006-08114) research project without his help. In addition, his friendship (including that of his family) and motivation were really appreciated and enjoyed for me. In all aspects, he was an incredible mentor.

In the same way, I would like to thank my parents' support and unconditional belief in me, not only for my PhD period of time but, during all my life. Also, the affection I have got and still get from my brothers, cousins, aunts and uncles makes me feel and remind them as if I were at home in a far away country.

I could not be grateful enough with my family -Meylí, my wife and Meylí Guadalupe†, Raúl Addí and Diego, my children- for their contribution in my PhD. They make my spirits high with their smiles and make me strong with their unquestioning love all along.

I would like to thank Prof. Carles Arús at UAB who allowed my collaboration with the GABRMN group through the European projects: e-Tumour (LSHC-CT-2004-503094) and HealthAgents (IST-2004-272 14). Also, he interceded for me with the INTERPRET European project (IST-1999-10310) partners (whom I am very grateful) to use the human brain tumour spectra dataset analysed in this work.
GABRMN group members were a remarkable set of collaborators, specially Margarida and Ana Paula, who provided their experience

and assistance for my adaptation not only in the group but in the Catalan culture as well.

I would like to thank the institutions which contributed to my PhD: SEP, UTM, UAB and UPC. The Secretary of Public Education of Mexico which provided my PhD grant through the PROMEP program. The Universidad Tecnológica de la Mixteca which provided the necessary facilities to get my PhD scholarship. The Universitat Autònoma de Barcelona, which let me collaborate with the GABRMN group at the Biochemical and Molecular Biology department. The Universitat Politècnica de Catalunya which allowed my collaboration with the SOCO group at the Languages and Computing department. In the same way, I would like to thank my thesis and papers anonymous reviewers for the extremely valuable feedback on my work.

Finally, I would like to thank all my unconditional friends, specially, at UPC and UTM during these last years. I really will miss some of my labs' friends as well as some of the UPC Mexican family. I will truly treasure some of the friendships I formed here in Barcelona.

Generative Manifold Learning for the Exploration of Partially Labeled Data

Raúl Cruz Barbosa

advisor

Alfredo Vellido

Departament de Llenguatges i Sistemes Informàtics

Universitat Politècnica de Catalunya

A thesis submitted for the degree of

Ph.D in Artificial Intelligence

In many real-world application problems, the availability of data labels for supervised learning is rather limited. Incompletely labeled datasets are common in many of the databases generated in some of the currently most active areas of research. It is often the case that a limited number of labeled cases is accompanied by a larger number of unlabeled ones. This is the setting for semi-supervised learning, in which unsupervised approaches assist the supervised problem and vice versa.

A manifold learning model, namely Generative Topographic Mapping (GTM), is the basis of the methods developed in this thesis. The non-linearity of the mapping that GTM generates makes it prone to trustworthiness and continuity errors that would reduce the faithfulness of the data representation, especially for datasets of convoluted geometry. In this thesis, a variant of GTM that uses a graph approximation to the geodesic metric is first defined. This model is capable of representing data of convoluted geometries. The standard GTM is here modified to prioritize neighbourhood relationships along the generated manifold. This is accomplished by penalizing the possible divergences between the Euclidean distances from the data points to the model prototypes and the corresponding geodesic distances along the manifold. The resulting Geodesic GTM (Geo-GTM) model is shown to improve the continuity and trustworthiness of the representation generated by the model, as well as to behave robustly in the presence of noise.

The thesis then leads towards the definition and development of semi-supervised versions of GTM for partially-labeled data exploration. As a first step in this direction, a two-stage clustering procedure that uses class information is presented. A class information-enriched variant of GTM, namely class-GTM, yields a first cluster description of the data. The number of clusters defined by GTM is usually large for visualization purposes and does not necessarily correspond to the overall class structure. Consequently, in a second stage, clusters are agglomerated using the K-means algorithm with different novel initialization strategies that benefit from the probabilistic definition of GTM. We evaluate if the use of class information influences cluster-wise class separability. A robust variant of GTM that detects outliers while effectively minimizing their negative impact in the clustering process is also assessed in this context.

We then proceed to the definition of a novel semi-supervised model, SS-Geo-GTM, that extends Geo-GTM to deal with semi-supervised problems. In SS-Geo-GTM, the model prototypes are linked by the nearest neighbour to the data manifold constructed by Geo-GTM. The resulting proximity graph is used as the basis for a class label propagation algorithm. The performance of SS-Geo-GTM is experimentally assessed, comparing positively with that of an Euclidean distance-based counterpart and that of the alternative Laplacian Eigenmaps method. Finally, the developed models (the two-stage clustering procedure and the semi-supervised models) are applied to the analysis of a human brain tumour dataset (obtained by Nuclear Magnetic Resonance Spectroscopy), where the tasks are, in turn, data clustering and survival prognostic modeling.

Contents

List of Figures

Part I

Background

Chapter 1

Introduction

Labeling aspects of reality seems to be one of the most standard occupations of the human brain and, therefore, of natural learning. When dividing the existing reality into different categories, we are seamlessly performing a classification task that can be improved over time through learning.

In the realm of non-natural, or machine learning, the task of unraveling the relationship between the observed data and their corresponding class labels can be seen as the modeling of the mapping between a set of data inputs and a set of discrete data targets. This is understood as supervised learning.

Unfortunately, in many real applications class labels are either completely or partially unavailable. The first case scenario is that of unsupervised learning, where the most common task to be performed is that of data clustering, which aims to discover the "true" group structure of multivariate data (Jain & Dubes, 1998). The second case is less frequently considered but far more common than what one might expect: quite often, only a reduced number of class labels is readily available and even that can be difficult and/or expensive to obtain.

One clear example of such situation is the type of data that will be the subject of a benchmark study in this thesis. The data in this study are spectra obtained through Nuclear Magnetic Resonance (NMR) spectroscopy, which are intended to provide the clinician (or the data analyst) with an accurate snapshot of the chemical composition of human brain tumour tissue samples. These spectra, by themselves, are difficult to obtain, standardize and preprocess for analysis (Tate *et al.*, 1998). On top of that, not all the tumour samples are likely to be correctly

classified (attributed to a tumour type) and many might not even be diagnosed at all.

In such context, unsupervised models are an adequate tool for a first exploratory approach. The available class labels can then be used to refine the unsupervised procedure. This becomes a task on the interface between supervised and unsupervised models: semi-supervised learning (Chapelle *et al.*, 2006). This type of learning is commonly understood as a way to improve supervised tasks (usually with few available labeled samples) with the use of unlabeled samples (Blum & Mitchell, 1998; Ghahramani & Jordan, 1994; Joachims, 1999; Nigam *et al.*, 2000; Seeger, 2000). In this thesis, the approach is a less typical one: improving and refining unsupervised learning by using class labeled data.

The baseline method we will resort to in order to follow this approach is a generative constrained mixture model of the manifold learning family: namely, Generative Topographic Mapping (GTM: Svensén 1998). This model has been quoted to be "a very powerful architecture in such situations, obtaining the latent manifold as a smooth nonlinear mapping of a uniform distribution over a low-dimensional space, represented by a regular grid" (Seeger, 2000). This regular grid low-dimensional representation allows GTM to be used for the intuitive visualization of both the multivariate data and the obtained clustering results.

The standard GTM, though, was defined to deal with data that could be represented reasonably well through low dimensional manifolds of smooth curvature (Svensén, 1998). Unfortunately, it has a limited capability to represent data of convoluted curvature. Another part of this thesis deals with this problem. In it we define a variation on GTM, namely the Geo-GTM, that replaces the commonly used Euclidean distance by a geodesic metric that favours data point similarities along the learned manifold. This alternative (dis)similarity measure can help to uncover the underlying structures of convoluted datasets, while still performing well in the modelling of datasets of smooth curvature.

1.1 Thesis' Goals and Contributions

In this section the main goals and the novel contributions of the thesis are summarily listed for the benefit of the reader. Amongst the main goals:

- The design and implementation of an extension of GTM that can face multivariate datasets with underlying convoluted geometric properties. The main task of the corresponding extension is the same as for standard GTM: clustering and visualization. It represents a first unsupervised exploratory approach and it will be the base for posterior semi-supervised models.

- The design and implementation of some semi-supervised methods based on the baseline manifold learning GTM model and its extensions. These semi-supervised methods aim to assist unsupervised data analysis strategies with the addition of class label information. All methods will be deployed within the framework of Statistical Machine Learning.

- The evaluation of the performance of these methods, as well as its comparison with the performance of alternative ones. In order to accomplish such evaluation, both adequate artificial and real datasets will be used.

- In the final version of this thesis, we plan to complete the application of these methods to a benchmark problem concerning the analysis of NMR spectra in a human oncology context. The obtained knowledge should become part of the outcome of the I+D+I TIN-2006-08114 research project of the Soft Computing group at the department of Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, which deals with the design and development of a decision support system for the assistance of clinicians in the diagnosis of human brain tumours.

Main theoretical novelties:

- Definition of a two-stage clustering procedure as a principled extension of GTM by explicitly using class-GTM in the first stage and K-means in the second one. Also, two novel initialization procedures for the second stage, derived from class-GTM training, are defined.

- Definition of Geo-GTM as a principled extension of GTM to uncover underlying structures in convoluted datasets, by explicitly penalizing the differences between the Euclidean and the alternative geodesic distance from data to prototypes in the original constrained mixture model.

- Definition of SS-Geo-GTM as a principled extension of Geo-GTM to semi-supervised problems, by explicitly introducing a modified label propagation algorithm on top of Geo-GTM.

Main expected application novelty:

- Novel application of semi-supervised models of the manifold learning family to the assistance of exploratory unsupervised clustering of real NMR spectrometric data with uncertain prognostic labeling, including the characterization of brain tumour typology and degree.

1.2 Thesis Overview

For an improved interpretability, the thesis contents are split into four parts. The rest of the thesis outline is structured as follows: In Part I, the necessary theoretical background is summarily reviewed. Chapter 2 provides a general overview of the semi-supervised learning problem in Pattern Recognition, aiming to stick to the viewpoint provided by the Statistical Machine Learning field. Out of the different modalities of semi-supervised learning, the focus will be placed in generative models. Chapter 3 is a self-contained summary of the basics of Generative Manifold Learning, including a more detailed description of the baseline model that will be used throughout the thesis: GTM.

The second part concerns data explorations without class information. In chapter 4, an unsupervised exploratory method dealing with datasets of convoluted geometric properties is presented. The basic idea is to uncover underlying structures in convoluted datasets. Part of the results presented in this chapter were published in Cruz-Barbosa & Vellido (2008a,b,d).

Part III deals with data explorations using class information. Chapter 5 presents an extension of GTM model using class information for clustering purposes. The early ideas for experimentation in this chapter were published in Cruz-Barbosa & Vellido (2006, 2007a,b,d), and applied to a preliminary human brain tumour characterization reflected in the final results of this chapter. Part of these results were published in Cruz-Barbosa & Vellido (2007c, 2008c).

In chapter 6, semi-supervised versions of GTM and Geo-GTM (the model developed in chapter 4) are described. Unlike in chapter 5, classification is here the main task these models are put to. Most of the contents of this chapter have been submitted for publication in Cruz-Barbosa & Vellido (2009c). Partial results, including the comparison of the proposed semi-supervised models with the alternative Laplacian Eigenmaps method were published in Cruz-Barbosa & Vellido (2009a). Also, a prognostic analysis (in a semi-supervised fashion) of a human brain tumour dataset using survival information as class labels has recently been submitted for publication in Cruz-Barbosa & Vellido (2009b). In the concluding part IV of the thesis, chapter 7 briefly summarizes the impact of the main contributions and provides a few pointers for future research.

Chapter 2

The Semi-Supervised Learning Problem in Pattern Recognition

2.1 Introduction

This chapter introduces some of the basic concepts underlying the field of semi-supervised learning, within the general framework of Machine Learning. It must be noted from the onset that this research area is still far from fully established and standardized, and that disparate approaches to deal with it can be found in the recent academic literature. In what follows, we shall stick to the view provided by the Statistical Machine Learning field.

Modern Pattern Recognition has for long been well served by Machine Learning techniques, many of them widely applied and accepted. There are many ways to categorize these techniques; amongst them, we are interested in that which divides them between supervised and unsupervised, according to the availability of data labels to accompany the data observations. It is common knowledge that, in supervised Machine Learning, the aim is to learn a mapping from the observed input data to an output whose correct values, or target labels, are provided by a supervisor. In unsupervised learning, instead, there is no such supervisor, and only unlabeled observed input data are available. The aim in this case is to find regularities that might exist in the input data.

Semi-Supervised Learning (SSL) is an emergent discipline that incorporates prior knowledge into supervised or unsupervised methods (classification and clus-

tering, mainly). The need for SSL, understood as learning from a combination of both labeled and unlabeled data, rises naturally in cases for which there exists a large supply of unlabeled data but a limited one of labeled data (bearing in mind that in many practical domains it can be very difficult and/or expensive to generate the labeled data). When SSL is used for classification, the main goal is to improve the classification accuracy aided by unlabeled data.

SSL for classification has become popular over the past few years. Some of the proposed methods include: co-training (Blum & Mitchell, 1998), in which there are two kinds (views) of information for training – about examples and the availability of both labeled and unlabeled data; Transductive Support Vector Machines (TSVM, Joachims 1999), in which transduction follows Vapnik's principle: when trying to solve some problems, one should not solve a more difficult problem as an intermediate step; and Expectation-Maximization (EM), within the Maximum Likelihood framework, to incorporate unlabeled data into the training processes (Ghahramani & Jordan, 1994; Nigam *et al.*, 2000).

In Seeger (2000) this task is defined as follows: Given an unknown probabilistic relationship $p(\mathbf{x}, y)$ between input points $\mathbf{x}$ and class labels $y \in Y = \{1, ..., c\}$, the problem is to predict y from $\mathbf{x}$, i.e. to find a *predictor* $\hat{y} = \hat{y}(x)$ such that the generalization error of $\hat{y}$,

$$p_{\mathbf{x},y}\{\hat{y}(\mathbf{x}) \neq y\}, \tag{2.1}$$

is small and ideally close to the *Bayes error*, being this the minimum of the generalization errors of all predictors. We are looking for algorithms to compute $\hat{y}$ from

- a labeled sample $D_l = \{(\mathbf{x}_i, y_i)|i = 1, ..., n\}$, where the $(\mathbf{x}_i, y_i)$ are drawn independently from $p(\mathbf{x}, y)$,

- an unlabeled sample $D_u = \{\mathbf{x}_i|i = n+1, ..., n+m\}$, where the $\mathbf{x}_i$ are drawn independently from the marginal input distribution $p(\mathbf{x}) = \sum_{y=1}^{c} p(\mathbf{x}, y)$. D_u is sampled independently from D_l.

- Prior knowledge (or assumptions) about the unknown relationship.

In unsupervised learning, one of the most widely used methods for data analysis is clustering. Clustering tries to group a set of points into clusters such that points in the same cluster are more similar to each other than to points in different clusters, under a particular cluster distortion or distance measure (Jain & Dubes, 1998).

Semi-supervised clustering (SSC) uses class labels or pairwise constraints (specifying wether two instances should be in same or different clusters) on some examples to aid unsupervised clustering. SSC is useful when knowledge of the relevant categories of a problem is incomplete. When it happens, SSC can group data using the categories in the initial labeled data as well as extend and modify the existing set of categories as needed to reflect other regularities in the data.

Two general approaches for SSC can be found in existing methods (Basu, 2005), namely: constraint-based and distance-based methods. In the former, the clustering algorithm itself is modified so that the available labels or constraints are used to bias the search for an appropiate clustering of the data. In the latter approaches, an existing clustering algorithm that uses a distance measure is employed; however, the distance measure is first trained to satisfy the labels or constraints in the supervised data.

At the present time, there is a tendency to consider as "standard" SSL methods (Chapelle *et al.*, 2006) only those which use it for classification tasks (as it is defined in Seeger 2000). However, SSC should be considered a more general SSL setting when the number and nature of the classes are not known in advance but have to be inferred from the data.

A problem related to SSL is transductive learning. Here a (labeled) training set and an (unlabeled) test set are provided. The idea of transduction is to perform predictions only for the test data.

2.2 Semi-Supervised Learning Categories

SSL methods work on the basis of some assumptions, which allow a general classification of the different techniques (Chapelle *et al.*, 2006):

- The semi-supervised smoothness assumption: if two points $\mathbf{x}_1, \mathbf{x}_2$ in a high density region are close, then so should be the corresponding outputs y_1, y_2. This assumption implies that if two points are separated by a low density region, then their outputs need not be close to each other.

- The cluster assumption: if points are in the same cluster, they are likely to be of the same class. This can be equivalently formulated as a low density separation criterion: the decision boundary should lie in a low-density region. Both formulations are conceptually equivalent but can inspire different algorithms.

- The manifold assumption: the (high-dimensional) data lie (roughly) on a low-dimensional manifold. This assumption allows to avoid the curse of dimensionality in the sense that when data happen to lie on a low-dimensional manifold, the learning algorithm can essentially operate in a space of corresponding dimension.

- Vapnik's principle: when trying to solve some problem, one should not solve a more difficult problem as an intermediate step. Transduction follows this principle, in this kind of problems as in supervised learning we want to predict a set of labels y corresponding to some objects $\mathbf{x}$. Transduction consists of directly estimating the finite set of test labels (a function $f : X_u\ (D_u) \rightarrow Y$ only defined on the test set) instead of inferring a function $f : X \rightarrow Y$ on the entire space X as in inductive methods.

Following the assumptions mentioned above, the SSL methods can be classified as (Chapelle *et al.*, 2006): generative models, low-density separation, graph-based methods and change of representation methods.

Inference in generative models involves the estimation of the conditional density $p(\mathbf{x}|y)$, where y represents class information. In this way, any additional information implicitly contained in the input data, reflected on $p(\mathbf{x})$, becomes useful. The cluster assumption is implemented using these models since a given cluster is assumed belong to only one class. Knowledge of the structure of the problem or the data can naturally be incorporated to the model (Nigam *et al.*,

2006). It is important to note, though, that unlabeled data can decrease prediction accuracy, when modeling assumptions are not correct (Cozman & Cohen, 2006).

The algorithms which try to implement the low-density separation assumption push the decision boundary away from the unlabeled points. To achieve this goal the most common method is Transductive Support Vector Machines, which try to implement transductive learning ideas (though some authors consider TSVM as a semi-supervised algorithm, see chapter 25 of Chapelle *et al.* 2006). The TSVM method maximizes the margin for unlabeled as well as for labeled points. Some alternatives to TSVM have been formulated in a probabilistic and in an information theoretic framework (Grandvalet & Bengio, 2006; Lawrence & Jordan, 2006).

In graph-based methods, the data are represented by the nodes of a graph, the edges of which are labeled with the pairwise distances of the incident nodes (and a missing edge corresponds to infinite distance). The way the distance between two points is computed can be seen as an approximation of the geodesic distance of the two points with respect to the manifold of data points (Belkin & Niyogi, 2004). Thus, the manifold assumption is the appropriate base to build graph methods. Usually some graph methods are transductive because of the prediction consists of labels for the unlabeled nodes, although recent work has extended graph-based methods to produce inductive solutions (Sindhwani *et al.*, 2006). Directed graphs used for information propagation have also been researched in this field (Burges & Platt, 2006).

Change of representation methods include algorithms that are not intrinsically semi-supervised, but instead perform two-step learning:

1. Perform an unsupervised step on all data, labeled and unlabeled, but ignoring the available labels.

2. Ignore the unlabeled data and perform plain supervised learning using the new distance, representation, or kernel built in step 1.

The semi-supervised smoothness assumption is implemented here since the representation is changed in such a way that small distances in high-density regions are conserved. Some graph-based methods are related to these algorithms

since the construction of the graph from the data can be seen as an unsupervised change of representation (Saul *et al.*, 2006; Zhu *et al.*, 2006).

2.3 Semi-Supervised Generative Models

2.3.1 Generative Models

As mentioned in the introduction, the main thrust of the thesis concerns generative baseline methods, which we now describe within the SSL framework.

The basic problem consists on modelling a probability density function $p(\mathbf{x})$, given a finite number of data points $X = \{\mathbf{x}_n\}_{n=1}^{N}$ drawn from that density function. Several approaches to face this problem stand out, including parametric, non-parametric and semi-parametric methods (Bishop, 1995).

In parametric methods, a specific functional form for the density model is assumed. The drawback of such an approach is that the particular form of parametric function chosen might be incapable of providing a good representation of the true density (model). Instead, no particular functional form is assumed in non-parametric methods, and the form of the density is determined entirely by the data. The problem in these methods is that the number of parameters in the model grows with the size of the data set.

The best of both approaches is merged in the semi-parametric estimation. Here, a very general class of functional forms is allowed, in which the number of adaptive parameters can be increased in a systematic way to build ever more flexible models, but where the total number of parameters in the model can be varied independently from the size of the data set.

The last approach is the one we are interested in. In particular, we focus on mixture of distributions models. In these models, the density function is formed from a linear combination of basis functions, where the number M of basis functions is treated as a parameter of the model and is typically much less than the number N of data points. Thus, the model for the density can be written as a linear combination of component densities $p(\mathbf{x}|j)$ in the form

$$p(\mathbf{x}) = \sum_{j=1}^{M} p(\mathbf{x}|j)p(j). \tag{2.2}$$

This representation is called a *mixture distribution* (Titterington *et al.*, 1985), McLachlan & Basford (1988) and the coefficients $p(j)$ are called the *mixing parameters*. The next constraints should be satisfied by $p(j)$ (which is the prior probability of the data point having been generated from component j of the mixture)

$$\sum_{j=1}^{M} p(j) = 1, \tag{2.3}$$

$$0 \leq p(j) \leq 1. \tag{2.4}$$

In the same way, the component density functions $p(\mathbf{x}|j)$ are normalized so that

$$\int p(\mathbf{x}|j)dx = 1. \tag{2.5}$$

To generate a data point from the probability distribution (2.2), one of the components j is first selected at random with probability $p(j)$, and then a data point is generated from the corresponding component density $p(\mathbf{x}|j)$.

The way $p(\mathbf{x}|j)$ is computed depends on the type of distributions chosen for the individual component densities. For example, if Gaussian distributions are selected, then we say we are working with a Gaussian mixture model and $p(\mathbf{x}|j)$ is computed as (assuming the Gaussians each have a covariance matrix $\sum_j = \sigma_j^2\mathbf{I}$, where $\mathbf{I}$ is the identity matrix, and a mean μ_j):

$$p(\mathbf{x}|j) = \frac{1}{(2\pi\sigma_j^2)^{d/2}} \exp\left\{-\frac{\|\mathbf{x} - \mu_j\|^2}{2\sigma_j^2}\right\} \tag{2.6}$$

A Maximum Likelihood approach is often used to determine the parameters of a (Gaussian or other) mixture model from a set of data. An elegant, practical and iterative procedure for estimating the mixture parameters is the Expectation-Maximization or EM algorithm (Dempster *et al.*, 1977).

These kind of generative models are the background for posterior chapters in which we will consider generative methods.

2.3.2 Semi-supervision in Generative Models

In this section we describe the way in which a generative model can be seen as a semi-supervised method.

We can find a description of how a generative method can be used for semi-supervised learning tasks in Seeger (2000), specially for classification ones. Within this context the class distributions $p(\mathbf{x}|y)$[1] can be modeled using model families $\{p(\mathbf{x}|y,\theta)\}$, where θ is a latent or hidden variable, and the class priors $p(y)$ by $\pi_y = p(y|\pi)$, $\pi = (\pi_y)_y$. An architecture of this type is referred to as a joint density model, since the full joint density $p(\mathbf{x}, y)$ is modeled by $\pi_y p(\mathbf{x}|y,\theta)$. For any fixed $\hat{\theta}$, $\hat{\pi}$, an estimate of $p(y|\mathbf{x})$ can be computed by Bayes' formula:

$$p(y|\mathbf{x},\hat{\theta},\hat{\pi}) = \frac{\hat{\pi}_y p(\mathbf{x}|y,\hat{\theta})}{\sum_{y'=1}^{M} \hat{\pi}_{y'} p(\mathbf{x}|y',\hat{\theta})}. \tag{2.7}$$

A model for the marginal $p(x)$ is

$$p(\mathbf{x}|\theta,\pi) = \sum_{y=1}^{M} \pi_y p(\mathbf{x}|y,\theta). \tag{2.8}$$

If labeled and unlabeled data are available, a natural criterion emerges as the *joint log likelihood* of both D_l and D_u[2],

$$\sum_{i=1}^{n} \log \pi_{y_i} p(\mathbf{x}_i|y_i,\theta) + \sum_{i=n+1}^{n+m} \log \sum_{y=1}^{M} \pi_y p(\mathbf{x}_i|y,\theta), \tag{2.9}$$

It is straightforward to consider this as an issue of Maximum Likelihood in the presence of missing data (treating y as a latent or unobservable variable[3]), which can in principle be tackled by the EM algorithm, or alternative methods such as direct gradient descent.

Limitations of generative techniques in SSL

In summary, generative techniques use a model family $\{p(\mathbf{x}, y|\theta,\pi)\}$ in order to model the joint data distribution $p(\mathbf{x}, y)$. These techniques use a mixture density estimation method for $p(\mathbf{x})$ on $X_l \cup X_u$ ($D_l \cup D_u$), treating y as a latent class variable, then using the labeled sample D_l in order to associate latent classes with actual ones. A problem with this approach is that the labeling provided by the

[1] y plays the role of j as in section 2.3.1

[2] D_l and D_u follow the corresponding definitions on section 2.1

[3] For a general description on latent variable models, the reader is referred to section 3.2.

unsupervised method may be inconsistent with D_l, in which case the clustering should be modified to achieve such consistency. Another problem when following the aforementioned strategy is that, for classification problems, generative methods might not always provide good solutions. That is, the maximization of the joint likelihood of a finite sample (for example) does not necessarily lead to a small classification error, because depending on the model it might be possible to make the likelihood increase more by improving the fit of $p(x)$ instead of that of $p(y|x)$. Some recent work describing these limitations can be found in Bouchard & Triggs (2004), Lasserre *et al.* (2006), Kaski *et al.* (2005), and Peltonen *et al.* (2004).

2. THE SEMI-SUPERVISED LEARNING PROBLEM IN PATTERN RECOGNITION

Chapter 3

Theoretical Foundations of Generative Manifold Learning

3.1 Introduction

The non-linear dimensionality reduction problem of manifold learning can be expressed as the recovery of meaningful low-dimensional structures hidden in high-dimensional data. This recovery should allow us to extract useful information and discover meaningful features, patterns and rules from data. This kind of techniques are used, amongst others, in the fields of data mining, knowledge management, engineering and retrieval, bioinformatics and neuroinformatics, decision support, signal processing, etc. As an example, let us think of a set of pixel images of an individual's face observed under different posing and lighting conditions; the manifold learning task would consist on the identification of the underlying face-characterizing variables (angle of elevation, direction of light, face-feature interdistances, etc.), given only the high-dimensional observed pixel image data (Tenenbaum *et al.*, 2000).

When the manifold assumption is taken up for clustering analysis, one important question is how to incorporate intrinsic geometric information of multivariate data in the corresponding clustering method. Identifying the underlying manifolds defining the data is of critical importance for their understanding. Usually, the methods used for finding embedded data structures look for global structures and/or local geometry.

Amongst the methods that identify global structures, mainly embedded linear subspaces, we find, for instance, Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). PCA (for general reference, see Jolliffe 2002, 2nd edition) is used to find the directions in which the variances are maximized. SVD (see, for instance, Golub & Reinsch 1970) finds the linear subspace that best preserves the information of the data. For both PCA and SVD, the constraint that the embedded structure must be globally linear is too restrictive for many applications (as it cannot capture nonlinear relationships defined by beyond second order statistics). Another method close to PCA and SVD is Multi-Dimensional Scaling (MDS or metric MDS). Metric MDS is used to map the high dimensional data into a low dimensional space, trying to preserve the inter-data distances (Young, 1981).

In methods that look for local geometry, the global linearity condition is usally left out. Recent years have witnessed the rapid development of nonlinear manifold methods. Four main approaches can be distinguished: The first one, based on projection methods, aims to find principal surfaces covering data-populated areas, such as principal curves (Hastie & Stuetzle, 1988; Kégl *et al.*, 2000).

The second entails local and global embedding algorithms. Amongst the former, Locally Linear Embedding (LLE, Roweis & Lawrence 2000) and Laplacian Eigenmaps (Belkin & Niyogi, 2003a), which focus on the local data neighbouring structure. Amongst the latter, Isometric Feature Mapping or ISOMAP (Tenenbaum *et al.*, 2000).

The third resorts to mutual information, which is a measurement of the differences of probability distribution between the observed and embedded spaces. Examples of these are Stochastic Nearest Neighbor (Hinton & Roweis, 2003) and Manifold Charting (Brand, 2003).

The fourth concerns generative models (GTM: Bishop *et al.* 1998), and hypothesizes that observed data are generated from a low-dimensional latent space.

Manifold learning models can also be considered according to the machine learning task they are fit for: supervised or unsupervised. In recent times, semi-supervised learning methods have made use of manifolds for classification tasks. Here, the fact that the data lie on a submanifold embedded in a high-dimensional

space, is assumed. In addition, learning algorithms developed under this assumption avoid the ubiquitous curse of dimensionality problem because they essentially operate in a space of corresponding (low) dimension.

In Chapelle *et al.* (2006), it is shown how several graph-based methods can be built under the manifold assumption. The main idea stemming for these methods is that the data are represented by the nodes of a graph (forming a manifold of data points) and the edges are labeled with the pairwise distances of the incident nodes. For example, in Belkin & Niyogi (2004) the approach is that classification functions are naturally defined only on the submanifold in question rather than the total ambient space. The problem with this approach is that a relatively small amount of noise or a few outliers can change the results dramatically. There are other approaches that take the problem in different directions (see Chapelle *et al.* 2006).

Not all generative models for manifold learning concern supervised learning (e.g. Bishop *et al.* 1998; de Silva & Tenenbaum 2003; Tenenbaum *et al.* 2000).

The unsupervised problem is stated as follows: Let Y be a d-dimensional domain contained in the Euclidean space $\mathbb{R}^d$, and let $f : Y \to \mathbb{R}^N$ be a smooth embedding, for some $N > d$. Data points $\{y_i\} \subset Y$ are generated by some random process, and are mapped by f to give the data observed, $\{x_i = f(y_i)\} \subset \mathbb{R}^N$. Y is referred as the latent space and $\{y_i\}$ as the latent data. The task is to reconstruct f and $\{y_i\}$ from the observed data $\{x_i\}$ alone. In the next section we describe the Generative Topographic Mapping, mentioned in previous chapters, as a model of this kind.

3.2 Generative Topographic Mapping

In this section we describe the Generative Topographic Mapping (GTM: Bishop *et al.* 1998; Svensén 1998). Since GTM is a latent variable model, we first provide a brief introduction to this kind of models.

Let $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ and $\mathbf{u} = (u_1, u_2, \ldots, u_q)$ be, in turn, a set of observable variables and a set of latent, hidden or unobservable variables. A latent variable model try to express the distribution $p(\mathbf{x})$ in terms of a smaller number of latent variables $\mathbf{u}$, where $q < d$ (Bishop, 1999). This can be achieved by decomposing

the joint distribution $p(\mathbf{x}, \mathbf{u})$ into the product of the marginal distribution $p(\mathbf{u})$ of the latent variables and the conditional distribution $p(\mathbf{x}|\mathbf{u})$ of the data variables given the latent variables, i.e. $p(\mathbf{x}, \mathbf{u}) = p(\mathbf{u})p(\mathbf{x}|\mathbf{u})$. The conditional distribution $p(\mathbf{x}|\mathbf{u})$ is expressed in terms of a mapping from latent variables to data variables in the following way: $\mathbf{x} = \mathbf{y}(\mathbf{u}; \mathbf{w}) + \beta$, where $\mathbf{y}(\mathbf{u}; \mathbf{w})$ is a function of the latent variable $\mathbf{u}$ with parameters $\mathbf{w}$, and β is an $\mathbf{u}$-independent noise process.

A latent variable model is said to be defined when the distribution $p(\beta)$, the mapping $\mathbf{y}(\mathbf{u}; \mathbf{w})$, and the marginal distribution $p(\mathbf{u})$ are specified. Geometrically the function $\mathbf{y}(\mathbf{u}; \mathbf{w})$ defines a manifold in data space given by the image of the latent space.

Finally, the desired model for the distribution $p(\mathbf{x})$ of the data is obtained by marginalizing over the latent variables $p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{u})p(\mathbf{u})d\mathbf{u}$.

The GTM is a generative non-linear latent variable model that, in its original definition, was intended for modelling continuous, intrinsically low-dimensional data distributions, embedded in high-dimensional spaces. It can also be understood both as a sound probabilistic alternative to the well-known and widely used Self-Organizing Maps (SOM: Kohonen 1995) and as a constrained mixture of distributions model. Its constraints make it less flexible than general mixtures of distributions, but such renounce to flexibility is compensated by computational expediency and by data visualization capabilities akin to those of the SOM, which general mixture models lack. Like SOM, GTM is used for unsupervised clustering and visualization.

3.2.1 The Standard GTM Model

The GTM is a non-linear latent variable model of the manifold learning family defined as a mapping from a low dimensional latent space onto the multivariate space where observed data reside. The mapping is carried through by a number of basis functions generating a constrained mixture density distribution. It is defined as a generalized linear regression model:

$$\mathbf{y}(\mathbf{u}; \mathbf{w}) = \phi(\mathbf{u})\mathbf{W} \tag{3.1}$$

where ϕ are M basis functions $\phi(\mathbf{u}) = (\phi_\mathbf{1}(\mathbf{u}), ..., \phi_\mathbf{M}(\mathbf{u}))$. For continuous data of dimension D, spherically symmetric Gaussians

$$\phi_m(\mathbf{u}) = \exp\left\{-1/2\sigma^2\|\mathbf{u} - \mu_m\|^2\right\} \tag{3.2}$$

are an obvious choice of basis function, with centres μ_m and common width σ; $\mathbf{W}$ is a $M \times D$ matrix of adaptive weights w_{md} that defines the mapping, and $\mathbf{u}$ is a point in latent space. To avoid computational intractability a regular grid of K points $\mathbf{u}_k$ can be sampled from the latent space. Each of them, which can be considered as the representative of a data cluster, has a fixed prior probability $p(\mathbf{u}_k) = 1/K$ and is mapped, using Eq. 3.1, into a low dimensional manifold non-linearly embedded in the data space. This latent space grid is similar in design and purpose to that of the visualization space of the SOM. A probability distribution for the multivariate data $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ can then be defined, leading to the following expression for the log-likelihood:

$$L(\mathbf{W}, \beta|\mathbf{X}) = \sum_{n=1}^{N} \ln\left\{\frac{1}{K}\sum_{k=1}^{K}\left(\frac{\beta}{2\pi}\right)^{D/2} \exp\left\{-\beta/2\|\mathbf{y}_k - \mathbf{x}_n\|^2\right\}\right\} \tag{3.3}$$

where $\mathbf{y}_k$, usually known as *reference* or *prototype vectors*, are obtained for each $\mathbf{u}_k$ using Eq. 3.1; and β is the inverse of the noise variance, which accounts for the fact that data points might not strictly lie on the low dimensional embedded manifold generated by the GTM.

The Expectation-Maximization (EM) algorithm (Dempster *et al.*, 1977) is an straightforward alternative to obtain the maximum likelihood estimates of the adaptive parameters of the model, which are the adaptive matrix of weights $\mathbf{W}$ and β (the EM algorithm for GTM is described in appendix A.1). In the E-step of the EM algorithm, the mapping is inverted and the responsibilities z_{kn} (the posterior probability of cluster k membership for each data point $\mathbf{x}_n$) can be directly computed as

$$z_{kn} = p(\mathbf{u}_k|\mathbf{x}_n, \mathbf{W}, \beta) = \frac{p(\mathbf{x}_n|\mathbf{u}_k, \mathbf{W}, \beta)p(\mathbf{u}_k)}{\sum_{k'} p(\mathbf{x}_n|\mathbf{u}_{k'}, \mathbf{W}, \beta)p(\mathbf{u}_{k'})}, \tag{3.4}$$

where $p(\mathbf{x}_n|\mathbf{u}_k, \mathbf{W}, \beta) = \mathcal{N}(\mathbf{y}(\mathbf{u}_k, \mathbf{W}), \beta)$.

3.2.2 Visualization using GTM

The interpretation of clustering results usually requires a drastic reduction of the dimensionality of the data. Latent variable models can provide such interpretation through visualization, as they describe the multivariate data in intrinsically low-dimensional spaces. The GTM was originally defined as an alternative to the SOM, defined within a probabilistic framework. As a result, the data visualization capabilities of the latter are fully preserved and even augmented by the former. The main advantage of GTM and any of its extensions over general finite mixture models consists precisely on the fact that both data and results can be intuitively visualized on a low dimensional representation space.

Each of the cluster representatives $\mathbf{u}_k$ in the latent visualization space is mapped, following Eq. 3.1, into a point $\mathbf{y}_k$ belonging to a manifold embedded in data space. Given that the posterior probability of every GTM cluster representative for being the generator of each data point $\mathbf{x}_n$ can be calculated, using Bayes' theorem, in the expectation step of the EM algorithm (Eq. 3.4), both data points and cluster prototypes can be visualized as a function of the latent point locations. The assignment of a probability of cluster membership to each data point n is a neat improvement on the SOM sharp map unit membership attribution for each data point, and leads to 2-dimensional representations of each multivariate data point in the form of the mean of the posterior distribution

$$u_n^{mean} = \sum_{k=1}^{K} \mathbf{u}_k z_{kn}, \tag{3.5}$$

or in the form of attributions to the latent space locations bearing maximum responsibility:

$$u_n^{maxresp} = \arg\max_{u_k} z_{kn}. \tag{3.6}$$

3.2.3 *t*-GTM

For the standard Gaussian GTM (described in section 3.2.1), the presence of outliers is likely to negatively bias the estimation of the adaptive parameters, distorting the clustering results. In order to overcome this limitation, the GTM was recently redefined (Vellido, 2006; Vellido & Lisboa, 2006) as a constrained

mixture of Student's t distributions: the t-GTM, aiming to increase the robustness of the model towards outliers. The mapping described by Equation 3.1 remains, with the basis functions now being Student's t distributions and leading to the definition of the following mixture density:

$$p(\mathbf{x}|\mathbf{W},\beta,\nu) = \frac{1}{K}\sum_{k=1}^{K}\frac{\Gamma(\frac{\nu+D}{2})\beta^{D/2}}{\Gamma(\frac{\nu}{2})(\nu\pi)^{D/2}}\left(1+\frac{\beta}{\nu}\|\mathbf{y}_k-\mathbf{x}\|^2\right)^{\frac{\nu+D}{2}} \tag{3.7}$$

where $\Gamma(\cdot)$ is the gamma function and the parameter $\nu = (\nu_1,\ldots,\nu_K)$ represents the degrees of freedom for each component k of the mixture, so that it can be viewed as a tuner that adapts the level of robustness (divergence from normality) for each component. This density leads to the redefinition of the model log-likelihood as

$$L(\mathbf{W},\beta,\nu|\mathbf{X}) = \sum_{n=1}^{N}\log\left\{\frac{1}{K}\sum_{k=1}^{K}\frac{\Gamma(\frac{\nu+D}{2})\beta^{D/2}}{\Gamma(\frac{\nu}{2})(\nu\pi)^{D/2}}\left(1+\frac{\beta}{\nu}\|\mathbf{y}_k-\mathbf{x}_n\|^2\right)^{\frac{\nu+D}{2}}\right\} \tag{3.8}$$

and, again, the estimation of the corresponding adaptive parameters is carried out by EM.

3.3 Distance Measure or Metric

Most manifold learning and non-linear dimensionality reduction methods use a specified metric to represent (dis)similarities between data points, and its definition is fundamental to the performance of their main tasks. The choice of metric or distance measure in a model such as GTM, which is based on the calculation of distances between data points and model prototypes, can be of critical importance to the performance of the method. Euclidean distances are one of the standard choices in mixture models such as GTM. Nevertheless, GTM is a constrained mixture in the sense that its defined prototypes are bound to lay in a manifold embedded in data space. GTM can therefore be considered a manifold learning model. In the process of non-linear mapping between the latent and data spaces, this manifold may fold in a way that makes the standard Euclidean distance a compromising choice. This becomes especially relevant in the analysis of multivariate datasets with convoluted geometries in the input data space.

A metric defines a distance between the items of a given data set. If X is such data set, then the corresponding distance function is $d : X \times X \to \mathbb{R}$, where $\mathbb{R}$ is the set of real numbers. This function is required to abide to the following conditions, for all x, y and z in X:

1. Non-negativity: $d(x, y) \geq 0$
2. Identity: $d(x, y) = 0$ if and only if $x = y$
3. Symmetry: $d(x, y) = d(y, x)$
4. Triangle inequality: $d(x, z) \leq d(x, y) + d(y, z)$

The Euclidean distance (or 2-norm distance) is widely-known to satisfy the conditions of a metric. Given two points $\mathbf{x} = (x_1, x_2, ..., x_n)$ and $\mathbf{y} = (y_1, y_2, ..., y_n)$ in $\mathbb{R}^n$ (Euclidean space), the Euclidean distance is defined as

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} |x_i - y_i|^2 \right)^{1/2}. \tag{3.9}$$

Most clustering algorithms use the Euclidean distance to find (dis)similarities between data points. However, this choice of distance may be problematic for the modelling of datasets of certain geometric properties. For example, in Fig. 3.1, it is obvious that a Euclidean distance-based method would assign the '?' data point to the 'x' class without hesitation, although an analyst might argue that, for reasons of geometric continuity, the data point should have been assigned to the 'o' class.

Since, as previously explained, the GTM defines a manifold embedded in the data space, we will surely face situations similar to the one sketched in Fig. 3.1. This problem might be alleviated by the use of an alternative metric. In this thesis, we propose an alternative, more adequate type of distance, thought to be more suitable for these situations: the Geodesic distance, which is described in section 4.2.

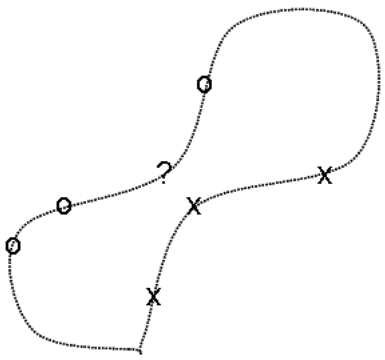

Figure 3.1: Two classes in a 2-D space. ‘o’ and ‘x’ are labeled examples, ‘?’ is a point to be classified. The remaining ‘.’ are unlabeled data points.

3. THEORETICAL FOUNDATIONS OF GENERATIVE MANIFOLD LEARNING

Part II

Exploration without class information

Chapter 4

Geodesic Generative Topographic Mapping

4.1 Introduction

The nonlinear dimensionality reduction (NLDR) methods belonging to the manifold learning family attempt to model high-dimensional multivariate data under the assumption that these can be faithfully represented by a low-dimensional manifold embedded in the observed data space. This simplifying assumption may, at worst, limit the faithfulness of the generated data mapping due to either data point neighbourhood relationships that do not hold in their low-dimensional representation, hampering its *continuity*, or spurious neighbouring relationships in the representation that do not have a correspondence in the observed space, which limit the *trustworthiness* of the low-dimensional representation. As described in the following sections, these concepts of *continuity* and *trustworthiness* can be transformed into operative metrics that will help us to qualify, without resorting to any element of subjectivity, how well the visualization (low-dimensional representation) obtained by dimensionality reduction represents the underlying data (Venna, 2007). These metrics find their motivation in the field of information retrieval, where the *trustworthiness* and *continuity* are related, in turn, to the concepts of *precision* and *recall*, of common use in machine learning as well.

The GTM (Bishop *et al.*, 1998), described in section 3.2, is a flexible manifold learning NLDR model for simultaneous data clustering and visualization whose

probabilistic nature makes possible to extend it to perform tasks such as missing data imputation (Vellido, 2006), robust handling of outliers and unsupervised feature selection (Vellido *et al.*, 2006), or time series analysis (Olier & Vellido, 2008), amongst others.

In the original formulation, GTM is optimized by minimization of an error that is a function of Euclidean distances, making it vulnerable to the aforementioned *continuity* and *trustworthiness* problems, especially for datasets of convoluted geometry. Such data may require plenty of folding from the GTM model, resulting in an unduly entangled embedded manifold that would hamper both the visualization of the data and the definition of clusters the model is meant to provide. Following an idea proposed in Archambeau & Verleysen (2005), the learning procedure of GTM is here modified by penalizing the divergences between the Euclidean distances from the data points to the model prototypes and the corresponding approximated geodesic distances along the manifold. By doing so, we prioritize neighbourhood relationships between points along the generated manifold, which makes the model more robust to the presence of off-manifold noise. In this chapter, we first assess to what extent the resulting Geodesic GTM (or Geo-GTM) model (which incorporates the data visualization capabilities that the model proposed in Archambeau & Verleysen 2005 lacks) is capable of preserving the *trustworthiness* and *continuity* of the mapping. Then we assess whether Geo-GTM shows better behaviour in the presence of noise than its standard GTM counterpart.

4.2 Manifolds and Geodesic Distances

As stated in the introduction, manifold learning methods work on the assumption that multivariate data can be faithfully represented by lower-dimensional manifolds embedded in the data space. Manifold methods such as ISOMAP (Tenenbaum *et al.*, 2000) and Curvilinear Distance Analysis (Lee *et al.*, 2002), for instance, use the geodesic distance as a basis for generating the data manifold. ISOMAP, in fact, can be seen as an instance of Multi-Dimensional Scaling (MDS) in which the Euclidean distance is replaced by the geodesic one. This metric measures similarity along the embedded manifold, instead of doing it through

the embedding space. In doing so, it may help to avoid some of the distortions (such as breaches of topology preservation) that the use of a standard metric such as the Euclidean distance may introduce when learning the manifold, due to its excessive folding (that is, undesired manifold curvature effects).

The otherwise computationally intractable geodesic metric can be approximated by graph distances (Bernstein *et al.*, 2000), so that instead of finding the minimum arc-length between two data points lying on a manifold, we would set to find the shortest path between them, where such path is built by connecting the closest successive data points. In this thesis, this is done using the K-rule, which allows connecting the K-nearest neighbors. A weighted graph is then constructed by using the data and the set of allowed connections. The data are the vertices, the allowed connections are the edges, and the edge labels are the Euclidean distances between the corresponding vertices. If the resulting graph is disconnected, some edges are added using a minimum spanning tree procedure in order to connect it. Finally, the distance matrix of the weighted undirected graph is obtained by repeatedly applying Dijkstra's algorithm (Dijkstra, 1959), which computes the shortest path between all data samples. See appendix B for more details about graph construction.

There exist alternative rules for building graphs (Lee & Verleysen, 2007). Amongst them, those solely based on the data set, such as the $\epsilon - rule$ (which defines a fixed ϵ-radius neighborhood criterion) or the $\tau - rule$ (a more complex local data density-adaptive neighborhood criterion), or those in which prototype-based models are used, such as the *Data-rule* and the *Histogram-rule* (Aupetit, 2003). A comparison between the performance of different graph rules goes, nevertheless, beyond the aims of our research.

4.3 Geo-GTM

The Geo-GTM model is an extension of GTM that favours the similarity of points along the learned manifold, while penalizing the similarity of points that are not contiguous in the manifold, even if close in terms of the Euclidean distance. This is achieved by modifying the standard calculation of the responsibilities in Eq. 3.4 proportionally to the discrepancy between the geodesic (approximated by

the graph) and the Euclidean distances. Such discrepancy is made operational through the definition of the exponential distribution, as in Archambeau & Verleysen (2005):

$$\mathcal{E}(d_g|d_e,\alpha) = \frac{1}{\alpha}\exp\left\{-\frac{d_g(\mathbf{x}_n,\mathbf{y}_m) - d_e(\mathbf{x}_n,\mathbf{y}_m)}{\alpha}\right\}, \tag{4.1}$$

where $d_e(\mathbf{x}_n,\mathbf{y}_m)$ and $d_g(\mathbf{x}_n,\mathbf{y}_m)$ are, in turn, the Euclidean and graph distances between data point $\mathbf{x}_n$ and the GTM prototype $\mathbf{y}_m$. Responsibilities are redefined as:

$$z_{mn}^{geo} = p(\mathbf{u}_m|\mathbf{x}_n,\mathbf{W},\beta) = \frac{p'(\mathbf{x}_n|\mathbf{u}_m,\mathbf{W},\beta)p(\mathbf{u}_m)}{\sum_{m'} p'(\mathbf{x}_n|\mathbf{u}_{m'},\mathbf{W},\beta)p(\mathbf{u}_{m'})}, \tag{4.2}$$

where

$$p'(\mathbf{x}_n|\mathbf{u}_m,\mathbf{W},\beta) = \mathcal{N}(\mathbf{y}(\mathbf{u}_m,\mathbf{W}),\beta)\mathcal{E}(d_g(\mathbf{x}_n,\mathbf{y}_m)^2|d_e(\mathbf{x}_n,\mathbf{y}_m)^2,1). \tag{4.3}$$

Here d_g and d_e are used as squared distances in order to be consistent with standard GTM (Bishop *et al.*, 1998). By setting α (the scale parameter of the distribution) to 1 in Eq. 4.1 we obtain the following required behaviour in Eq. 4.2. When d_g and d_e distances are similar the new responsibility, z_{mn}^{geo}, is almost not adjusted (the penalizing factor, Eq. 4.1, approaches to 1) then behaving as z_{mn} in standard GTM. However, when there is no agreement between the graph approximation of the geodesic distance and the Euclidean distance, the value of the numerator of the fraction within the exponential in Eq. 4.1 increases, pushing the modified responsibility in Eq. 4.2 towards smaller values, i.e., automatically punishing the discrepancy between metrics. In other words, we use the exponential distribution (Eq. 4.1) as a penalizing factor in Eq. 4.3 to follow the behaviour previously described for z_{mn}^{geo}. Once the responsibility is calculated in the modified E-step, the rest of the model's parameters are estimated following the standard EM procedure (as in appendix A.1).

The only additional computational effort incurred by Geo-GTM is the result of building a graph and the computation of its corresponding distance matrix, which is calculated only once before the EM algorithm is run. The dominant computational complexity of Geo-GTM is therefore similar to that of GTM.

4.3.1 Data Visualization using Geo-GTM

As for standard GTM, each of the cluster representatives $\mathbf{u}_m$ in the latent visualization space is mapped, following Eq. 3.1, into a point $\mathbf{y}_m$ (the center of a mixture component) belonging to a manifold embedded in data space. It is this mapping (and the possibility to invert it, defined by the responsibilities in Eq. 4.2) what provides Geo-GTM with the data visualization capabilities that the alternative Manifold Finite Gaussian Mixtures model proposed in Archambeau & Verleysen (2005) lacks. Given that the posterior probability of every Geo-GTM cluster representative for being the generator of each data point, or responsibility z_{mn}^{geo}, is calculated as part of the modified EM algorithm, data points can, once again, be visualized as a function of the latent point locations as the mean of the estimated posterior distribution:

$$\mathbf{u}_n^{mean} = \sum_{m=1}^{M} \mathbf{u}_m z_{mn}^{geo}, \tag{4.4}$$

or in the form of attributions to the latent space locations bearing maximum responsibility:

$$\mathbf{u}_n^{maxresp} = \arg\max_{\mathbf{u}_m} z_{mn}^{geo} \ . \tag{4.5}$$

4.4 Experiments

Geo-GTM was implemented in MATLAB®. For the experiments reported next, the adaptive matrix $\mathbf{W}$ was initialized, following a procedure described in Bishop *et al.* (1998), as to minimize the difference between the prototype vectors $\mathbf{y}_m$ and the vectors that would be generated in data space by a partial Principal Component Analysis (PCA). The inverse variance β was initialised to be the inverse of the 3^{rd} PCA eigenvalue. The initialization of $\mathbf{W}$ and β using a PCA-based procedure ensures the replicability of the results. The latent grid was fixed to a square layout of approximately $(N/2)^{1/2} \times (N/2)^{1/2}$, where N is the number of points in the dataset. The corresponding grid of basis functions was equally fixed to a 5×5 square layout for all datasets.

The goal of the experiments is threefold. Firstly, we aim to assess whether the proposed Geo-GTM model could capture and visually represent the underlying

structure of datasets of smooth but convoluted geometry better than the standard GTM. Secondly, we aim to quantify the faithfulness of the generated mappings. Finally, we aim to evaluate the capability of Geo-GTM to uncover the underlying structure of the data in the presence of noise, and compare its performance with that of the standard GTM.

4.4.1 Results and Discussion

Three artificial 3-dimensional datasets, represented in Fig. 4.1, were used in the experiments that follow. The first one is *Swiss-Roll*, consisting on 1000 randomly sampled data points generated by the function: $(x_1, x_2, x_3) = (t\cos(t), t_2, t\sin(t))$, where t and t_2 follow uniform distributions $\mathcal{U}(3\pi/2, 9\pi/2)$ and $\mathcal{U}(0, 21)$, respectively. The second dataset, herein called *Two-Spirals*, consists of two groups of 300 data points each that are similar to *Swiss-Roll* although, this time, the first group follows the uniform distribution $\mathcal{U}(3\pi/4, 9\pi/4)$, while the second group was obtained by rotating the first one by 180 degrees in the plane defined by the first and third axes and translating it by 2 units along the resulting third axis. The third dataset, herein called *Helix*, consists of 500 data points that are images of the function $\mathbf{x} = (\sin(4\pi t), \cos(4\pi t), 6t - 0.5)$, where t follows $\mathcal{U}(-1, 1)$. These data are contaminated with a small level of noise. Also, and specifically for the experiments to assess the way the models deal with the presence of noise, Gaussian noise of zero mean and increasing standard deviation, from $\sigma = 0.1$ to $\sigma = 0.5$, was added to a noise-free version of *Helix* to produce the 5 datasets represented in Fig. 4.2.

The posterior mean distribution visualization maps for all datasets are displayed in Figs. 4.3 to 4.5. Geo-GTM, in Fig. 4.3, is shown to capture the spiral structure of *Swiss-Roll* far better than standard GTM, which misses it at large and generates a poor data visualization with large overlapping between non-contiguous areas of the data.

A similar situation is reflected in Fig. 4.4: The two segments of *Two-Spirals* are neatly separated by Geo-GTM, whereas the standard GTM suffers a lack of contiguity of the segment represented by circles as well as overlapping of part of the data of both segments.

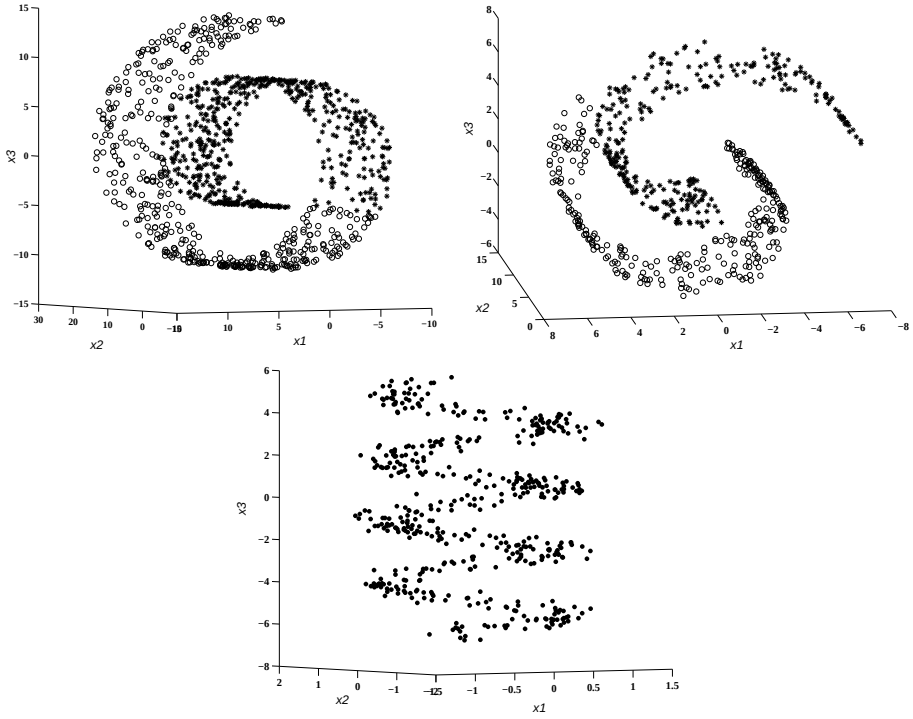

Figure 4.1: The three datasets used in the experiments. (Top-left): *Swiss-Roll*, where two contiguous fragments are identified with different symbols in order to check manifold contiguity preservation in Fig. 4.3. (Top-right): *Two-Spirals*, again with different symbols for each of the spiral fragments. (Bottom): *Helix*.

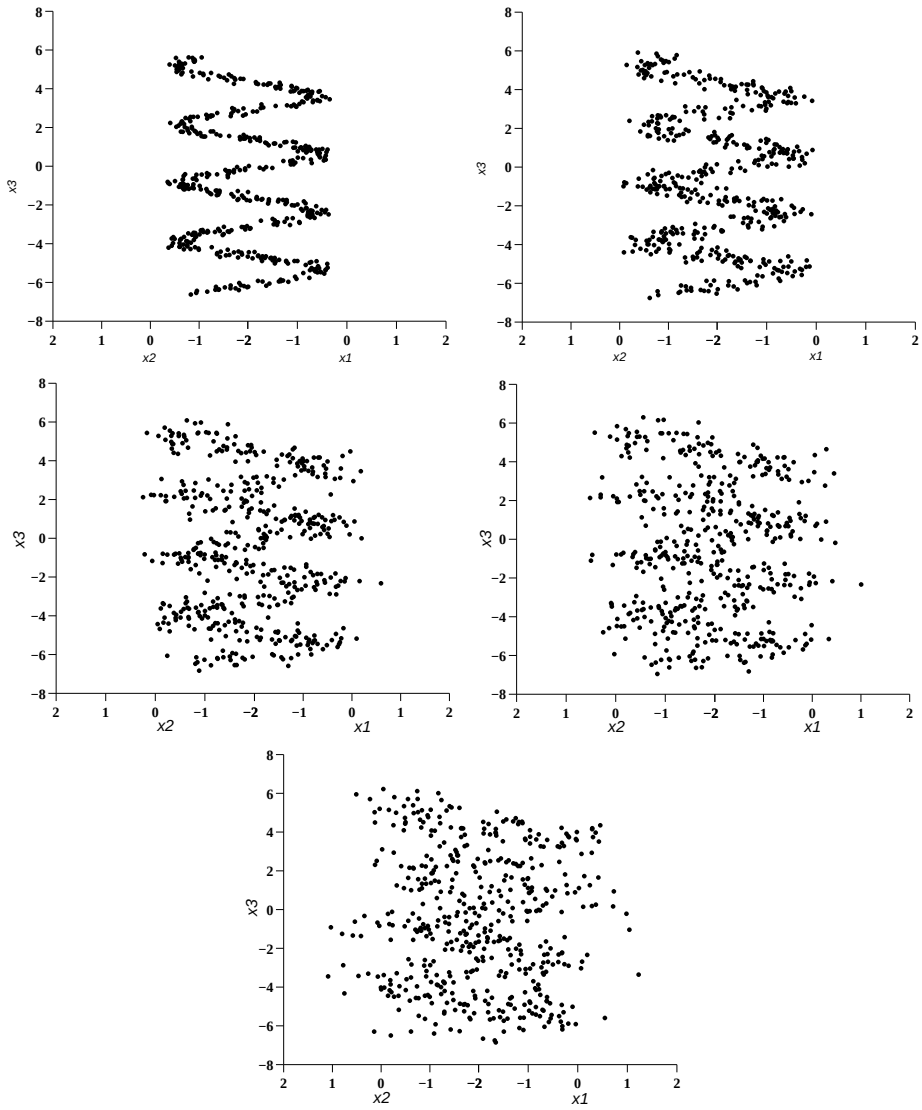

Figure 4.2: The five noisy variations of *Helix* used in the experiments. From left to right and top to bottom, with increasing noise of standard deviation from $\sigma = 0.1$ to $\sigma = 0.5$.

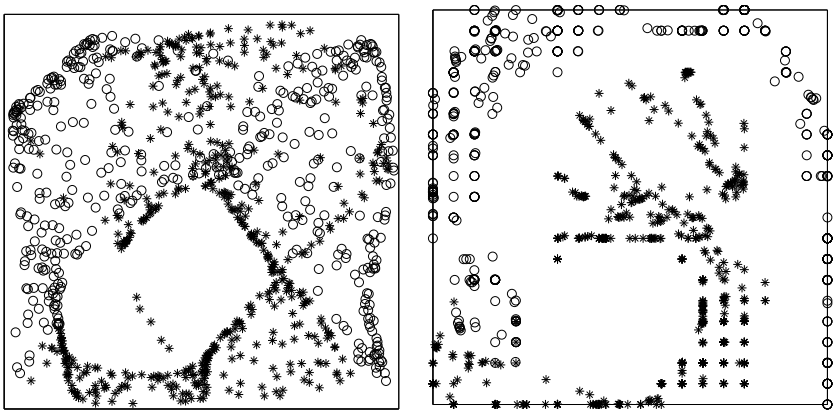

Figure 4.3: Data visualization maps for the *Swiss-Roll* set. (Left): standard GTM; (right): Geo-GTM. The axes of these latent spaces correspond to the components of the latent vectors of the model. They are, therefore, meaningless and, for this reason, they remain unlabeled. The same holds for figures 4.4 and 4.5.

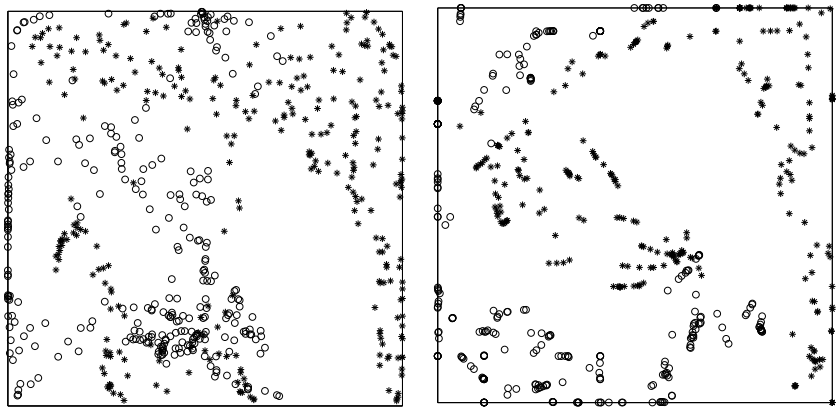

Figure 4.4: Visualization maps for the *Two-Spirals* set. (Left): standard GTM; (right): Geo-GTM.

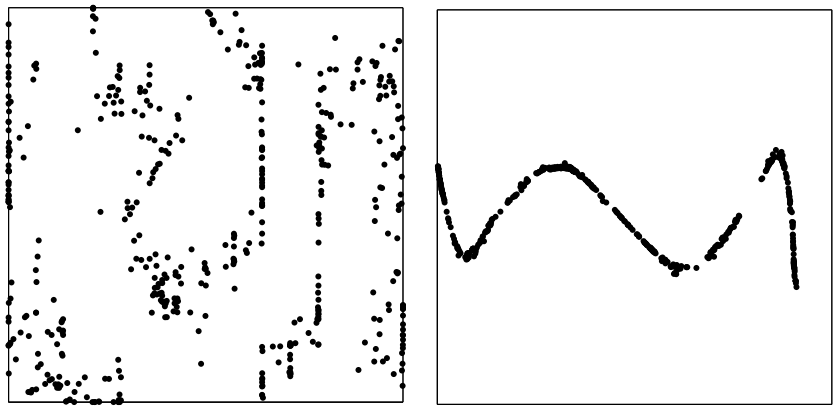

Figure 4.5: Data visualization maps for the *Helix* set. (Left): standard GTM; (right): Geo-GTM.

The results are even more striking for *Helix*, as shown in Fig. 4.5: the helicoidal structure is neatly revealed by Geo-GTM, whereas it is mostly missed by the standard GTM. The former also faithfully preserves data continuity, in comparison to the breaches of continuity that hinder the visualization generated by the latter.

In order to evaluate and compare the mappings generated by GTM and Geo-GTM, we use the *trustworthiness* and *continuity* measures developed in Venna & Kaski (2001). *Trustworthiness* is defined as:

$$T(K) = 1 - \frac{2}{NK(2N - 3K - 1)} \sum_{i=1}^{N} \sum_{x_j \in U_K(x_i)} (r(x_i, x_j) - K), \qquad (4.6)$$

where $U_k(x_i)$ is the set of data points x_j for which $x_j \in \hat{C}_K(x_i) \wedge x_j \notin C_K(x_i)$ and $C_K(x_i)$ and $\hat{C}_K(x_i)$ are the sets of K data points that are closest to x_i in the observed data space and in the low-dimensional representation space, respectively. In other words, it measures the error when data points that are not neighbours in the input space can be mapped close-by in the output space causing data points to be falsely identified as neighbours. Expressed in information retrieval terms, this kind of error decreases the *precision* of the visualization (low-dimensional

representation). *Continuity* is in turn defined as:

$$Cont(K) = 1 - \frac{2}{NK(2N - 3K - 1)} \sum_{i=1}^{N} \sum_{x_j \in V_K(x_i)} (\hat{r}(x_i, x_j) - K), \tag{4.7}$$

where $V_K(x_i)$ is the set of data points x_j for which $x_j \notin \hat{C}_K(x_i) \wedge x_j \in C_K(x_i)$. The terms $r(x_i, x_j)$ and $\hat{r}(x_i, x_j)$ are the ranks of x_j when data points are ordered according to their distance from the data vector x_i in the observed data space and in the low-dimensional representation space, respectively, for $i \neq j$. Here, it measures the error when data points that are originally close-by are mapped far away in the representation space. This kind of error is reflected by discontinuities in the mapping. In information retrieval terms, it reduces *recall*.

The measurements of *trustworthiness* and *continuity* for all datasets are shown in Fig. 4.6. As expected from the visualization maps in Figs. 4.3-4.5, the Geo-GTM mappings are far more trustworthy than those generated by GTM for neighbourhoods of any size across the analyzed range. The differences in continuity preservation are smaller although, overall, Geo-GTM performs better than GTM model, specially with the noisier *Helix* dataset.

We finally evaluate, through some illustrative experiments, the capability of Geo-GTM to uncover the underlying structure of the data in the presence of noise, comparing it with that of the standard GTM. We quantify it using the log-likelihood (Eq. 3.3), as applied to a test dataset consisting of 500 randomly sampled data points from a noise-free version of *Helix*. For further testing, we repeat the experiment with noisy variations of a basic dataset, herein called *Two-Helix* consisting of two sub-groups of 300 data points each, which are, in turn, images of the functions $\mathbf{x}1 = (sin(4\pi t), cos(4\pi t), 6t - 0.5)$ and $\mathbf{x}2 = (-sin(4\pi t), -cos(4\pi t), 6t - 0.5)$, where t follows $\mathcal{U}(-1, 1)$. This is, in fact, a DNA-like shaped duplication of the *Helix* dataset. The corresponding results are shown in Fig. 4.7.

Remarkably, Geo-GTM is much less affected by noise than the standard GTM, as it recovers with much higher likelihood the underlying noise-free functions. This corroborates the visualization results reported in Fig. 4.5, in which the standard GTM generates a far less faithful representation of the underlying form

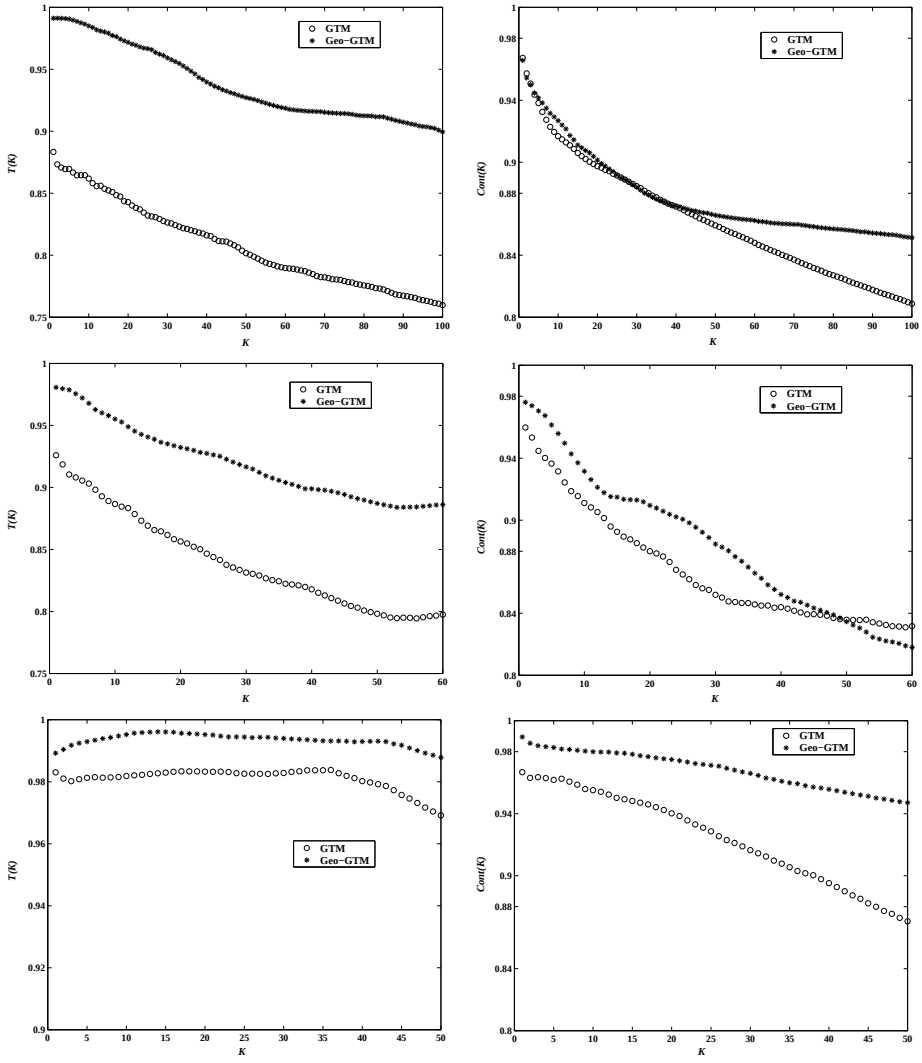

Figure 4.6: Trustworthiness (left column) and continuity (right column) for (top row): *Swiss-Roll*, (middle row): *Two-Spirals*, and (bottom row): *Helix*, as a function of the neighbourhood size K.

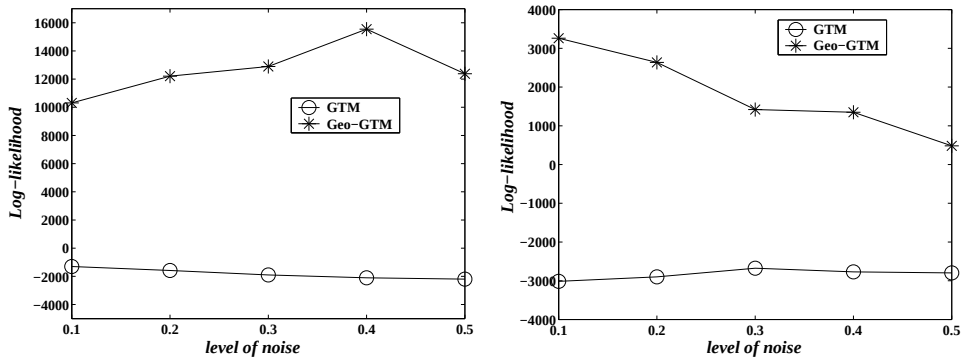

Figure 4.7: Test log-likelihood results for the *Helix* (left) and *Two-Helix* (right) datasets, for increasing levels of added uninformative noise.

and with breaches of continuity. This is probably due to the fact that Geo-GTM favours directions along the manifold, minimizing the impact of off-manifold noise.

4.5 Summary

In this chapter, we have introduced a variation of the NLDR manifold learning GTM model, namely Geo-GTM, which limits the effect of manifold folding through the penalization of the discrepancies between inter-point Euclidean distances and the approximation of geodesic distances along the model manifold. Through several experiments, evaluated by trustworthiness and continuity measures, it has been shown to faithfully recover and visually represent the underlying structure of datasets of smooth but convoluted geometries. The reported experiments also show that Geo-GTM behaves more robustly than the standard GTM in the presence of a considerable amount of noise in the datasets. The proposed model is the base of several developments in next chapters of this thesis.

A limitation of the proposed model is that it assumes an intrinsic continuity of the data (so that *Continuity* itself is a measure of performance). This, obviously, does not necessarily hold for data that are, even if partially, multi-modal. It has to be noted, though, that GTM and all its variants, as developed in this thesis, allow for both the direct visual and quantitative assessment of how multi-modality affects data representation. This can be achieved by calculating and

plotting together the mean (Eq. 4.4) and the mode (Eq. 4.5) for each data point and comparing them.

Part III

Explorations with class information

In Chapter 4, we have defined an evaluated a variant of GTM that favours the similarity of points along the learned manifold, increasing the faithfulness of the data representation. In Part III of this thesis, we take a step further to also account for the available class information. In Chapter 5, a novel two-stage clustering model, based on GTM and its variants, is improved by the use of class information. In Chapter 6, the Geo-GTM model presented in Chapter 4 is extended to a semi-supervised setting: the resulting SS-Geo-GTM is intended to assist classification tasks.

Chapter 5

Two-stage Clustering with class-GTM

5.1 Introduction

Class information is most commonly used for supervised classification problems. The use of class information in unsupervised clustering, instead, is a far less frequently investigated problem. This is the starting point for the current chapter, in which class labels are used to enrich and refine the cluster structure discovered in a two-stage clustering process.

Amongst density-based methods, Finite Mixture Models have established themselves as a flexible and robust tool for multivariate data clustering (Figueiredo & Jain, 2002). In many practical data analysis scenarios, though, the available knowledge concerning the cluster structure of the data may be quite limited. In these cases, data exploration techniques are valuable tools and, amongst them, multivariate data visualization can be of great help by providing the analyst with intuitive cues about data structural patterns. In order to endow Finite Mixture Models with data visualization capabilities, certain constraints must be enforced. One alternative is forcing the model components to be centred in a low-dimensional manifold embedded into the usually high-dimensional observed data space. Such approach is the basis of the definition of GTM, which has been introduced in section 3.2.

Finite Mixture Models can also be used beyond unsupervised learning in order to account for class-related information in supervised or semi-supervised settings (Hastie & Tibshirani, 1996). Class information can be integrated as part of the GTM training to enrich the cluster structure definition provided by the model (Cruz-Barbosa & Vellido, 2006; Sun *et al.*, 2002). The resulting class-GTM model is the basis of this chapter.

GTM in general and class-GTM in particular do not place any strong restriction on the number of mixture components (or clusters), in order to achieve an appropriate visualization of the data. This richly detailed cluster structure, which exploits the substructure in the input data, does not necessarily match the more global cluster and class distributions of the data. For that reason, a two-stage clustering procedure may be useful in this scenario (Vesanto & Alhoniemi, 2000). Class-GTM can be used in the first stage to generate a detailed cluster partition in the form of a mixture of components. The centres of these components, also known as prototypes, can be further clustered in the second stage. For that role, the well-known K-means algorithm is used in this study. The issue remains of how we should initialize K-means in the second clustering stage. Random initialization, with the subsequent choice of the best solution, was the method selected in Vesanto & Alhoniemi (2000). This approach, though, does not make use of the prior knowledge generated in the first stage of the procedure. Here, we propose two different ways of introducing such prior knowledge in the initialization of the second stage K-means, without compromising the final clusterwise class separation capabilities of the model. This fixed initialization procedures allow significant computational savings.

5.2 Two-Stage Clustering

The two-stage clustering procedure outlined in the introduction is described in this section. The first stage model, namely class-GTM, is introduced first. This is followed by the details of different initialization strategies for the second stage. We propose two novel second stage fixed initialization strategies that take advantage of the prior knowledge obtained in the first stage.

5.2.1 The class-GTM Model

In model-based clustering, we attempt to discover the cluster structure of the multivariate data $\mathbf{X}$ through the modelling of their distribution $p(\mathbf{X})$. In classification problems, instead, for which class labels are available for all data records, we attempt to model the relationship between these labels t and the data, in the form of the class probability $p(t|\mathbf{X})$. It is not unusual to find that classes overlap to a varying extent; that is, finding class distributions that do not correspond completely to the cluster distribution of the data. These two approaches are not necessarily incompatible, and we can aim to discover the cluster structure of the data while taking into account the available class information in a semi-supervised approach. This can be achieved by explicitly modelling the joint density $p(t, \mathbf{X})$.

The class-GTM model is an extension of GTM and therefore inherits most of its properties. The main goal of this extension is to improve class separability in the clustering results of GTM. For this purpose, we assume that the clustering model accounted for the available class information. This can be achieved by modelling the joint density $p(t, \mathbf{X})$, instead of $p(\mathbf{X})$, for a given set of classes $\{T_i\}$. For the Gaussian version of the GTM model (Cruz-Barbosa & Vellido, 2006; Sun *et al.*, 2002), such approach entails the calculation of the posterior probability of a cluster representative $\mathbf{u}_k$ given the data point $\mathbf{x}_n$ and its corresponding class label t_n, or class-conditional *responsibility* $z^t_{kn} = p(\mathbf{u}_k|\mathbf{x}_n, t_n)$, as part of the E step of the EM algorithm. It can be calculated as:

$$z^t_{kn} = \frac{p(\mathbf{x}_n, t_n|\mathbf{u}_k)}{\sum_{k'=1}^{K} p(\mathbf{x}_n, t_n|\mathbf{u}_{k'})} = \frac{p(\mathbf{x}_n|\mathbf{u}_k)p(t_n|\mathbf{u}_k)}{\sum_{k'=1}^{K} p(\mathbf{x}_n|\mathbf{u}_{k'})p(t_n|\mathbf{u}_{k'})} = \frac{p(\mathbf{x}_n|\mathbf{u}_k)p(\mathbf{u}_k|t_n)}{\sum_{k'=1}^{K} p(\mathbf{x}_n|\mathbf{u}_{k'})p(\mathbf{u}_{k'}|t_n)}, \tag{5.1}$$

and, being T_i each class,

$$p(\mathbf{u}_k|T_i) = \frac{\sum_{n;t_n=T_i} p(\mathbf{x}_n|\mathbf{u}_k) / \sum_n p(\mathbf{x}_n|\mathbf{u}_k)}{\sum_{k'} \sum_{n;t_n=T_i} p(\mathbf{x}_n|\mathbf{u}_{k'}) / \sum_n p(\mathbf{x}_n|\mathbf{u}_{k'})} \quad . \tag{5.2}$$

Equation 5.1 differs from the standard responsibility z_{kn} of GTM in that, instead of imposing a fixed prior $p(\mathbf{u}_k) = 1/K$ on latent space, we consider a class-conditional prior $p(\mathbf{u}_k|T_i)$. Once the class-conditional responsibility is calculated,

the rest of the model's parameters are estimated following the standard EM procedure.

In a similar way a class-t-GTM model is obtained, but the corresponding $p(\mathbf{x}_n|\mathbf{u}_k)$ is defined as

$$p(\mathbf{x}_n|\mathbf{u}_k) = \frac{\Gamma(\frac{\nu_k+D}{2})\beta^{D/2}}{\Gamma(\frac{\nu_k}{2})(\nu_k\pi)^{D/2}}\left(1+\frac{\beta}{\nu_k}\|\mathbf{y}_k-\mathbf{x}\|^2\right)^{\frac{\nu_k+D}{2}} . \tag{5.3}$$

5.2.2 Two-stage Clustering Based on GTM

In the first stage of the proposed two-stage clustering procedure, a class-GTM is trained to obtain the representative prototypes (detailed clustering) of the observed dataset $\mathbf{X}$. As mentioned in the introduction, the number of prototype vectors is usually chosen to be large for visualization purposes, and does not necessarily reflect the global cluster and class structure of the data. In this study, the resulting prototypes $\mathbf{y}_k$ of the class-GTM are further clustered using the well-known K-means algorithm (a description of which can be found, for instance, in Duda *et al.* 2000). In a similar two-stage procedure to the one described in Vesanto & Alhoniemi (2000), based on SOM, the second stage K-means initialization in this study is first randomly replicated 100 times, subsequently choosing the best available result, which is the one that minimizes the error function

$$E = \sum_{c=1}^{C}\sum_{\mathbf{x}\in G_c}\|\mathbf{x}-\mu_c\|^2, \tag{5.4}$$

where C is the final number of clusters in the second stage and μ_c is the centre of the K-means cluster G_c. This approach seems somehow wasteful, though, as the use of GTM instead of SOM can provide us with richer a priori information to be used for fixing the K-means initialization in the second stage.

Two novel fixed initialization strategies that take advantage of the prior knowledge obtained by class-GTM in the first stage are proposed. They are based on two features of the model, namely: the Magnification Factors (MF, Bishop *et al.* 1997) and the Cumulative Responsibility (CR, Vellido *et al.* 2000). The MF measure the level of stretching that the mapping undergoes from the latent to the data spaces. Areas of low data concentration correspond to high distorsions of

the mapping (i.e., high MF), whereas areas of high data density correspond to low MF. The MF is described in terms of the derivatives of the basis functions $\phi_j(\mathbf{u})$ in the form:

$$MF = \frac{dA'}{dA} = \det{}^{1/2}\left(\psi^{\mathbf{T}}\mathbf{W}^{\mathbf{T}}\mathbf{W}\psi\right), \tag{5.5}$$

where ψ has elements $\psi_{ji} = \partial\phi_j/\partial u^i$ (Bishop *et al.*, 1997). If we choose C to be the final number of clusters for K-means in the second stage, the first proposed fixed initialization strategy will consist on the selection of the class-GTM prototypes corresponding to the C non-contiguous latent points with lowest MF for K-means initialization. That way, the second stage algorithm is meant to start from the areas of highest data density.

As its name suggests, the CR is the sum of responsibilities over all data points in $\mathbf{X}$ for each cluster k:

$$CR_k = \sum_{n=1}^{N} z_{kn}^t \quad . \tag{5.6}$$

The second proposed fixed initialization strategy, based on CR, is similar in spirit to that based on MF. Again, if we choose C to be the final number of clusters for K-means in the second stage, the fixed initialization strategy will now consist on the selection of the class-GTM prototypes corresponding to the C non-contiguous latent points with highest CR. That is, the second stage algorithm is meant to start from those cluster prototypes that are found to be most responsible for the generation of the observed data.

5.3 Experiments with Publicly Available Datasets

In this section, we first describe the experimental design and settings. Two publicly available data sets, described in some detail in section 5.3.2 will be used, namely *e-coli*[1] and *oil-flow*[2]. This is followed by a presentation and discussion of the corresponding results.

[1] http://archive.ics.uci.edu/ml/

[2] http://research.microsoft.com/∼cmbishop/PRML/webdatasets/datasets.htm

5.3.1 Experimental Design and Settings

The class-GTM model was implemented in MATLAB®. For the experiments reported next, the adaptive matrix $\mathbf{W}$ was initialized, following a procedure described in Bishop *et al.* (1998), as to minimize the difference between the prototype vectors y_k and the vectors that would be generated in data space by a partial PCA, $m_k = V_2 u_k$, where the columns of matrix V_2 are the two principal eigenvectors (given that the latent space considered here is 2-dimensional). Correspondingly, the inverse variance β was initialised to be the inverse of the 3^{rd} PCA eigenvalue. This ensures the replicability of the results. The value of parameter σ, describing the common width of the basis functions, was set to 1. The grid of latent points $\mathbf{u}_k$ was fixed to a square 13×13 layout for the *e-coli* dataset and to a 20×20 layout for the *oil-flow* dataset. Both datasets are summarily described in section 5.3.2. The corresponding grid of basis functions ϕ was equally fixed to a 5×5 square layout for both datasets.

The goals of these experiments are twofold. First, we aim to assess whether a two-stage clustering procedure, where the first stage involves class-GTM and the second stage involves K-means, improves on the class separation capabilities of the straight clustering of the data using the K-means algorithm alone. Secondly, we aim to test whether the second stage initialization procedures based on the Magnification Factors and the Cumulative Responsibility of the class-GTM, described in section 5.2.2, retain the class separability capabilities of the two-stage clustering procedure in which K-means is randomly initialized. If this is the case, a fixed second stage initialization strategy should entail a substantial reduction of computational time compared to a random second stage initialization requiring a large number (100 in the reported experiments and also in Vesanto & Alhoniemi 2000) of algorithm runs.

Beyond the visual exploration that could be provided by class-GTM, the second stage clustering results should be explicitly quantified in terms of class separability. For that purpose, the following entropy-like measure is proposed:

$$E_{G_c}(\{T_i\}) = -\sum_{\{G_c\}} P(G_c) \sum_{\{T_i\}} P(T_i|G_c) \ln P(T_i|G_c) = -\sum_{c=1}^{C} \frac{K_{G_c}}{K} \sum_{i=1}^{|\{T_i\}|} p_{ci} \ln p_{ci} \quad . \tag{5.7}$$

Sums are performed over the set of classes $\{T_i\}$ and the K-means clusters $\{G_c\}$; K is the total number of prototypes; K_{G_c} is the number of prototypes assigned to the c^{th} cluster; $p_{ci} = K_{G_c i}/K_{G_c}$, where $K_{G_c i}$ is the number of prototypes from class i assigned to cluster c; and, finally, $|\{T_i\}|$ is the cardinality of the set of classes. The minimum possible entropy value is 0, which corresponds to the case of no clusters being assigned prototypes corresponding to more than one class.

Given that the use of a second stage in the clustering procedure is intended to provide final clusters that best reflect the overall structure of the data, the problem remains of what is the most adequate number of clusters. This is a time-honoured matter of debate, which goes beyond the scope of this thesis, and many cluster validity indices have been defined over the years. In this experiment we use the widely known Davies-Bouldin (DB) index (Davies & Bouldin, 1979; Vesanto & Alhoniemi, 2000) to provide us with some indication of what the adequate number of final clusters might be. According to the DB index, the best clustering minimizes

$$\frac{1}{C}\sum_{k=1}^{C}\max_{l\neq k}\left\{\frac{S_c(G_k)+S_c(G_l)}{d_{ce}(G_k,G_l)}\right\}, \tag{5.8}$$

where C is the number of clusters; S_c is a within-cluster distance named centroid distance and is calculated as $S_c = \frac{\sum_{\mathbf{y}_i\in G_k}\|\mathbf{y}_i-\mu_k\|}{N_k}$, N_k is the number of samples in cluster G_k, and μ_k is the center or mean of cluster G_k; and d_{ce} is a between-clusters distance named centroid linkage defined as $d_{ce}(G_k,G_l) = \|\mu_k - \mu_l\|$.

5.3.2 Results and Discussion

In the first stage of the two-stage clustering procedure, class-GTM was trained to model the *e-coli* and *oil-flow* datasets. The resulting prototypes $\mathbf{y}_k$ were then clustered in the second stage using the K-means algorithm. This last stage was performed in three different ways, as described in section 5.2.2. In the first one, K-means was randomly initialized 100 times, selecting the results corresponding to the minimum of the error function in Eq. 5.4. In the second, we used the Magnification Factors of class-GTM as prior knowledge for the initialization of K-means. In the third, Cumulative Responsibility was used as prior knowledge.

In all cases, K-means was forced to yield a given number of final clusters, from 2 up to 13. The DB index and the final entropy were calculated for all the above procedures and numbers of clusters.

The DB index results for the experiments with e-coli, including the direct clustering of the data with K-means alone, are reported in Fig. 5.1. E-coli consists of 336 7-dimensional points belonging to 8 classes representing protein location sites, 3 of which are very small, i.e., the data set is strongly class-unbalanced. The sensibility of the DB-index to class-unbalance might perhaps explain why the results in Fig. 5.1 do not provide a clear pattern.

The DB index results for the experiments with *e-coli*, including the direct clustering of the data with K-means alone, are reported in Fig. 5.1. *E-coli* consists of 336 7-dimensional points belonging to 8 classes representing protein location sites, 3 of which are very small, i.e., the data set is strongly class-unbalanced. The sensibility of the DB-index to class-unbalance might perhaps explain why the results in Fig. 5.1 do not provide a clear pattern. They nevertheless suggest that no more of 4 clusters (for two-stage clustering) or 5 (for direct K-means) represent an adequate solution. In fact, there are only 4 main groups in *e-coli*, namely: cytoplasm, periplasm, inner membrane and outer membrane. Some relatively good solutions are also suggested for 8 or 9 clusters using the two-stage procedure.

The entropy results for *e-coli* are shown in Fig. 5.2. Two immediate conclusions can be drawn: First, all the two-stage clustering procedures based on class-GTM perform much better than direct K-means clustering in terms of class separation in the resulting clusters. Second, random initialization in the second stage of the clustering procedure does not entail any significant advantage over the proposed fixed initialization strategies across the whole range of possible final number of clusters, while being far more costly in computational terms.

The DB index results for the experiments with *oil-flow*, also including the direct clustering of the data with K-means, are reported in Fig. 5.3. *Oil-flow*, firstly used in Bishop & James (1993), simulate non-intrusive measurements by gamma densitometry from a pipeline transporting a mixture of gas, oil, and water. It consists of 1000 points described by 12 attributes. Three types of flow configuration are used as class information labels. The results in Fig. 5.3 do not

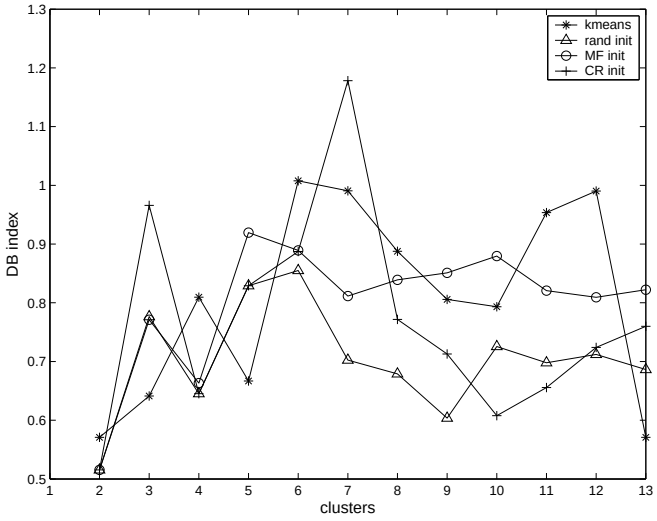

Figure 5.1: DB index for the clustering of *e-coli* using two-stage clustering with different initializations (based on Magnification Factors (MF init), Cumulative Responsibility (CR init) and random (rand init)), and K-means alone.

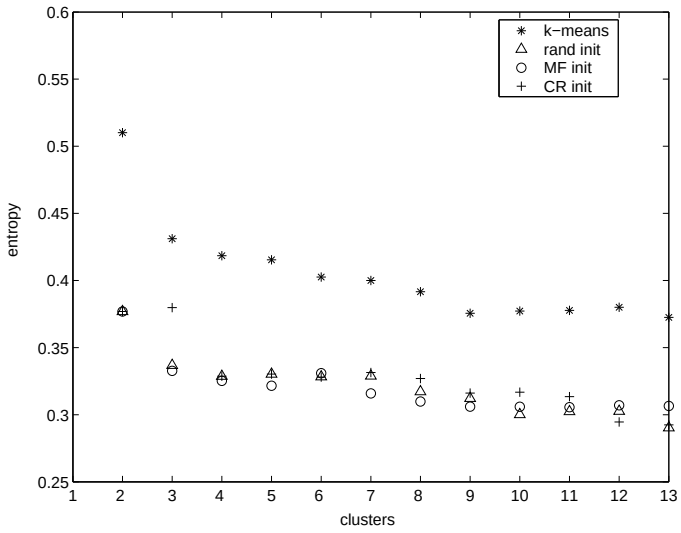

Figure 5.2: Entropy measurements for two stage and K-means alone clusterings of *e-coli*. Legend as in Fig. 5.1.

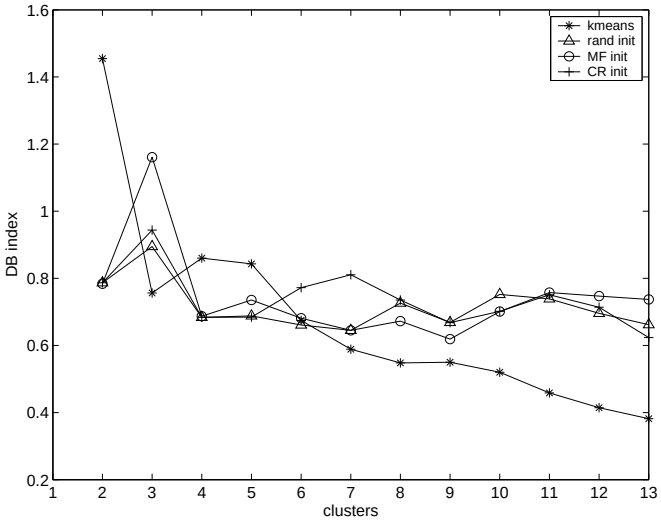

Figure 5.3: DB index for the clustering of *oil-flow* using two-stage clustering with different initializations and K-means alone. Legend as in Fig. 5.1.

indicate any clear number of clusters when data are grouped directly by K-means without any class information. Instead, for the two-stage procedure based on class-GTM there is no indication that more than 4 clusters would provide any substantial improvement.

The entropy results for *oil-flow* are shown in Fig. 5.4 and they are fully consistent with the results for *e-coli*. Again, the two-stage clustering procedures based on class-GTM perform much better than direct K-means clustering in terms of class separation, and the two-stage random and fixed initialization strategies yield almost identical results, with the former being computationally more costly.

5.4 Experiments on a Human Brain Tumour Dataset

Magnetic Resonance Spectroscopy (MRS) is a non-invasive tool capable of providing a detailed fingerprint of the biochemistry of living tissue. The data used in this study consist of 304 single voxel PROBE (PROton Brain Exam system) spec-

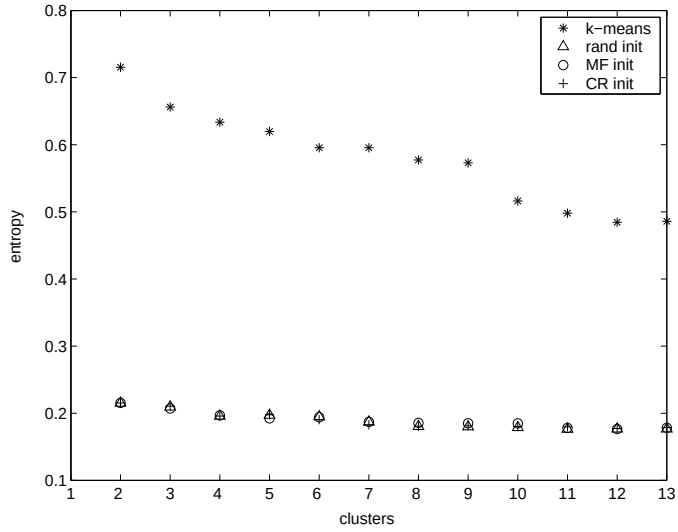

Figure 5.4: Entropy measurements for two stage and K-means alone clusterings of *oil-flow*. Legend as in Fig. 5.1.

tra acquired in vivo for fourteen viable tumour types: meningiomas (58 cases), glioblastomas (86), metastases (38), astrocytomas of 2 (22) and 3 (7) grades, PNETs (9), oligoastrocytomas (6), oligodendrogliomas (7), rare (19), pilocytic astrocytoma (3), malignant lymphomas (10), haemangioblastomas (5), abscesses (8), and schwannomas (4); as well as from adjacent normal brain tissue (22). This typology will be used in this study as class information. A description of the automated protocol used for the acquisition of these data can be found in Tate *et al.* (2003). The clinically relevant regions of the spectra were sampled to obtain 200 frequency intensity values. For a second interesting analysis, the spectra can be grouped into three types (typology that will be used in this study as class information), as in Tate *et al.* (2006): high grade malignant (metastases and glioblastomas), low grade gliomas (astrocytomas, oligodendrogliomas and oligoastrocytomas) and meningiomas. These groups will be considered as a second dataset in section 5.4.2. The complexity of the problem, in terms of high dimensionality, was compounded by the small number of spectra available, which is commonplace in MRS data analysis (Lisboa *et al.*, 2000). This makes either clustering or visualization almost compulsory for automated data analysis.

5.4.1 Experimental Design and Settings

For the experiments reported next, the adaptive matrix $\mathbf{W}$, the inverse variance β and the value of parameter σ were initialized as in section 5.3.1. The grid of latent points $\mathbf{u}_k$ was fixed to a square 12×12 layout. The corresponding grid of basis functions ϕ was equally fixed to a 5×5 square layout.

This time, the goals of these experiments are fourfold. First, we aim to assess whether the inclusion of class information using class-GTM in the first stage of the two-stage procedure results in any improvement in terms of clusterwise class separability (and under what circumstances) compared to the procedure using standard GTM. Second, as in the previous experiments, we aim to assess whether the two-stage procedure improves, in the same terms, on the use of direct clustering of the data using K-means. Third, and again as in the previous experiments, we aim to test whether the second stage initialization procedures based on MF and CR of the class-GTM, described in section 5.2.2, retain the class separability capabilities of the two-stage clustering procedure in which K-means is randomly initialized. If this is the case, a fixed second stage initialization strategy should entail a substantial reduction of computational time compared to a random second stage initialization. In fourth place, we aim to explore the properties of the structure of the dataset concerning atypical data. For that, we use a variant of the GTM (with and without class information) that behaves robustly in the presence of outliers, which was described in section 3.2.3: the t-GTM (Vellido, 2006).

The MRS data, described in section 5.4, will be first clustered using both GTM and class-GTM to illustrate the differences between these models. The results will be first compared visually, which should help to illustrate the visualization capabilities of the models. Beyond the visual exploration that could be provided by class-GTM and GTM, the second stage clustering results should be explicitly quantified in terms of clusterwise class separability. For that purpose, the proposed entropy-like measure in Eq. 5.7 is used.

In this experiment we do not use any cluster validity index and we simply evaluate the entropy measure for solutions from 2 up to 15 clusters.

5.4.2 Results and Discussion

In the first stage of the two-stage clustering procedure, GTM, t-GTM and their class-enriched variants class-GTM and class-t-GTM were trained to model the human brain tumour dataset described in section 5.4. The resulting prototypes $\mathbf{y}_k$ were then clustered in the second stage using the K-means algorithm. This last stage was performed in three different ways, as described in section 5.2.2. In the first one, K-means was randomly initialized 100 times, selecting the results corresponding to the minimum of the error function in Eq. 5.4. In the second, we used the Magnification Factors of class-GTM as prior knowledge for the initialization of K-means. In the third, Cumulative Responsibility was used as prior knowledge. In all cases, K-means was forced to yield a given number of final clusters, from 2 up to 15 (and from 2 up to 10 for the second dataset). The final entropy was calculated for all the above procedures and numbers of clusters.

Before considering the entropy results, visualization maps (obtained using the mean of the posterior distribution: $\sum_{k=1}^{K} \mathbf{u}_k z_{kn}$ or $\sum_{k=1}^{K} \mathbf{u}_k z_{kn}^t$) of all the trained models in the first stage were generated. Three hypotheses are made for the clustering results visualized here. First, the use of class information in the clustering models should yield visualization maps where the classes are separated better than in those models which do not use it. Second, the use of t-GTM should help to diminish the influence of outliers. Consequently, the visualization maps generated with these models should show the data more homogeneously distributed throughout the visualization maps than in Gaussian GTM models which do no use it. Thirdly, since the tumour dataset is mainly compound of poorly represented classes, we hypothesize that these "small" classes will consist mainly of atypical data.

The clustering model proposed to test the second and third hypothesis is t-GTM, a variant of the standard GTM that replaces the mixture of Gaussians by a mixture of Student's t-distributions, which are known to be best at dealing with atypical data, given their heavier tails. Details on the formulation of t-GTM can be found in Vellido (2006) (also a brief summary is presented in section 3.2.3). The two-stage clustering experiments were repeated for t-GTM

without class information and for class-t-GTM (Cruz-Barbosa & Vellido, 2006), the corresponding variant of the model with class information.

Given the complexity of the entire dataset, we only provide one of these illustrative visualizations in Fig. 5.5. Here, two tumour types (meningioma and glioblastoma, the most represented classes) are shown. The right column of Fig. 5.5, where the models that include class information are located, suggests that the first hypothesis is sustained, since the class separability between both classes ('o' and '+') is better than that of the models that do not make use of class information, located in the left column. This is the result of a more pronounced overlapping of both classes, clearly seen in the left hand-side models of Fig. 5.5.

The use of t-distributions in the models represented in the bottom row is more spread throughout the map than that of the Gaussian models of the top row. This is an indication that the t-GTM models are moderating the effect of outliers. The differences are not huge and, again, this is an indication that there might be not too many outliers in the dataset. All the previous results were generally supported for the rest of the data as well. The two first hypotheses are, therefore, preliminarily supported.

A similar situation can be appreciated in Fig. 5.6 for the second dataset (described in section 5.4), where two tumour groups (low grade gliomas and meningiomas) are shown, although the differences in class overlapping are less obvious in this case.

We now turn our attention to the third hypothesis. It was shown in Vellido & Lisboa (2006) that a given data instance could be characterized as an outlier if the value of

$$O_n^* = \sum_k z_{kn}\beta\|\mathbf{y}_k - \mathbf{x}_n\|^2 \tag{5.9}$$

was sufficiently large. Here, the right-side elements of Eq. 5.9 are as in (t-GTM) section 3.2.3. The histogram in Fig. 5.7 displays the values of O_n^* from Eq. 5.9 for the entire brain tumour dataset. First of all, and supporting our previous impression, not too many data could be clearly characterized as outliers according to this histogram. We did the same for the class-t-GTM model and, for illustration, proposed an artificial threshold. The 20 largest values of O_n^* were taken as outliers. The results are summarized in Table 5.1. Surprisingly,

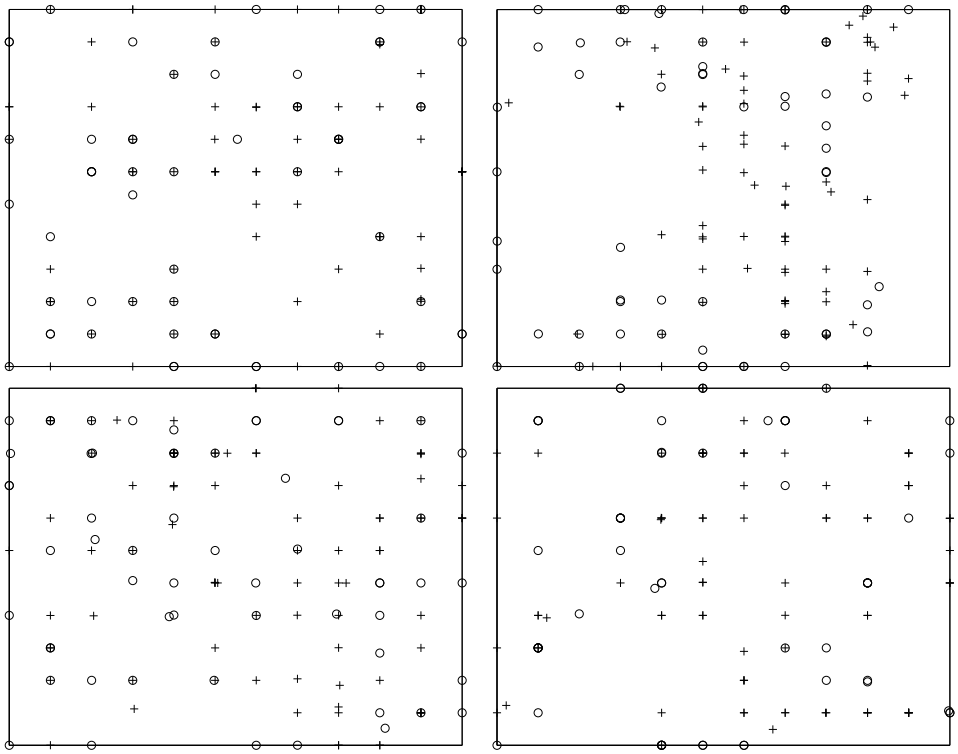

Figure 5.5: Representation, on the 2-dimensional latent space of GTM and its variants, of part of the entire tumour data set described in the main text. The representation is based on the mean posterior distributions for the data points belonging to meningioma ('o') and glioblastoma ('+') tumour types. The axes of the plot are the elements of the latent vector $\mathbf{u}$ and convey no meaning by themselves. For that reason, axes are kept unlabeled. (Top left): GTM without class information. (Top right): class-GTM. (Bottom left): t-GTM without class information. (Bottom right): class-t-GTM.

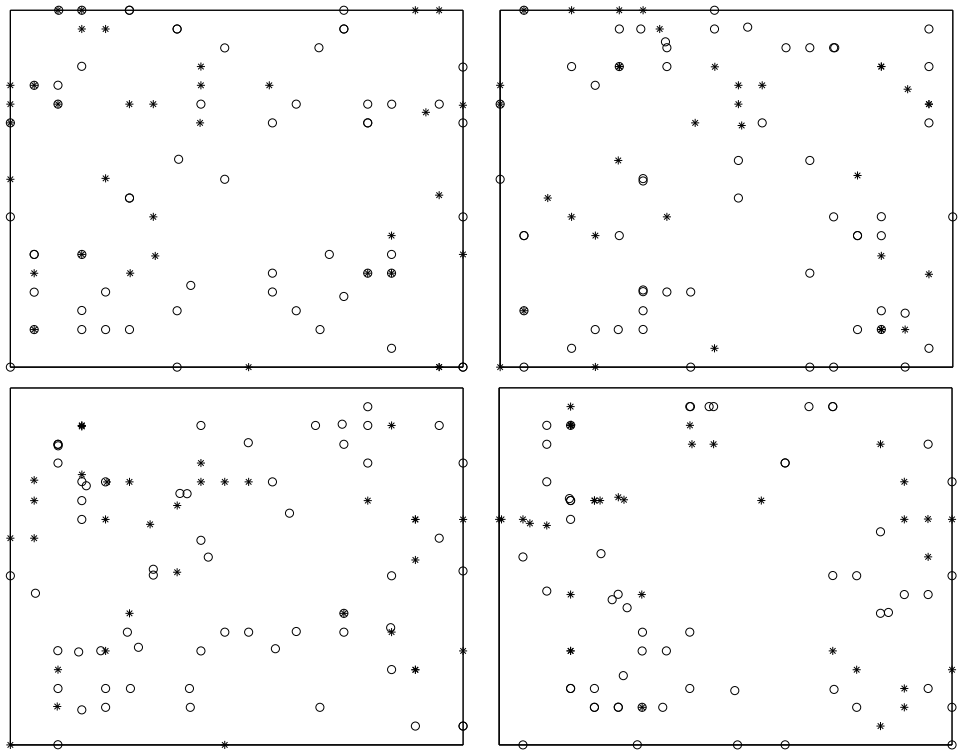

Figure 5.6: Representation, on the 2-dimensional latent space of GTM and its variants, of a part of the second tumour dataset. It is based on the mean posterior distributions for the data points belonging to low grade gliomas ('*') and meningiomas ('o'). The axes of the plot convey no meaning by themselves and are kept unlabeled. (Top left): GTM without class information. (Top right): class-GTM. (Bottom left): t-GTM without class information. (Bottom right): class-t-GTM.

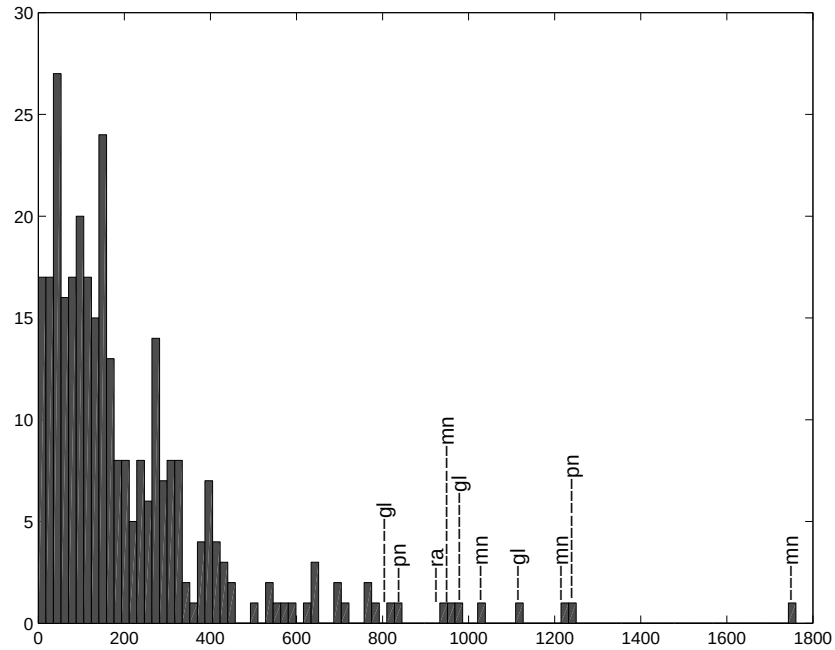

Figure 5.7: Histogram of the statistic (Eq. 5.9); outliers are characterized by its large values. For illustration, the ten largest values are labeled. See tumour type acronyms in Table 5.1.

Tumour type	# of outliers (%): t-GTM	# of outliers (%): class-t-GTM
Meningioma (*mn*)	6 (10.3%)	4 (6.9%)
Glioblastoma (*gl*)	6 (7.0%)	5 (5.8%)
Metastases (*me*)	1 (2.6%)	2 (5.3%)
Astrocytoma 2 (*a2*)	1 (4.5%)	4 (18.2%)
PNET (*pn*)	2 (22.2%)	2 (22.2%)
Rare (*ra*)	2 (10.5%)	2 (10.5%)
Lymphoma (*ly*)	1 (10.0%)	0 (0.0%)
Haemanglioblastoma (*hb*)	1 (20.0%)	1 (20.0%)

Table 5.1: Outlier count and percentage (in brackets) by tumour type (see figures in section 5.4) given a threshold for Eq. 5.9.

given the complex tumour typology of the dataset, these results do not support the third hypothesis, as many outliers belong to the tumour types with better representation in the dataset (*mn*, *gl*, *me*, and *a2*).

The histograms in Figs. 5.8 and 5.9 reflect similar results for the second dataset.

The entropy measurements quantifying the clusterwise class separation for the entire brain tumour dataset are shown in Fig. 5.10. Two immediate conclusions can be drawn. Firstly, all the two-stage clustering procedures based on class-GTM perform much better than the direct clustering of the data through K-means, in terms of class separation, but also better than the two-stage procedure without class information based on the standard GTM. Also, this situation is reflected in right hand side of Fig. 5.11 for the second dataset. However, it can also be observed in Fig. 5.11 that the two-stage clustering based on t-GTM performs slightly better than the class-t-GTM model. This is explained by the fact that the adjustment of the model provided by t-GTM, which is blind to class information by itself, may alter the accordance between class and cluster distributions, especially in a strongly class-unbalanced dataset such as the one under analysis. This result draws the limits out of which the addition of class information is not necessarily useful in terms of cluster-wise separation. Secondly, in both datasets, random initialization in the second stage of the clustering procedure, with or without class information, does not entail any significant advantage over

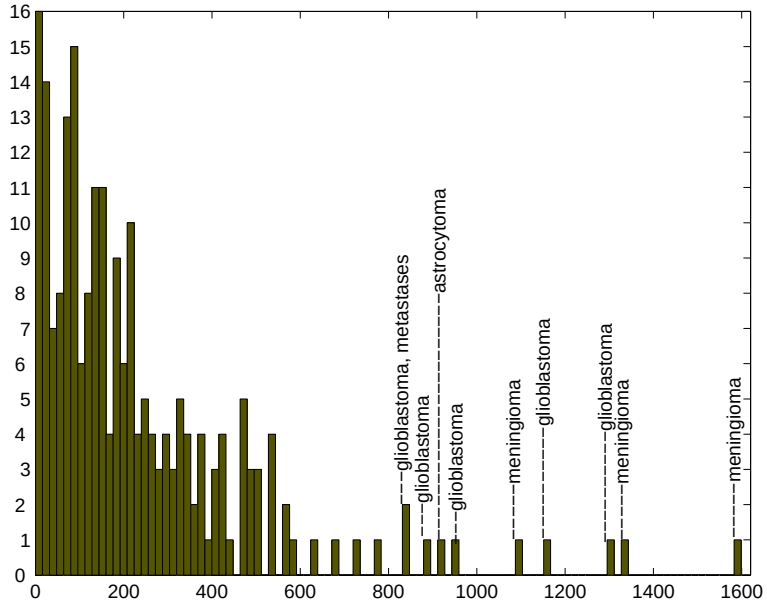

Figure 5.8: Histogram of the statistic (Eq. 5.9) for the t-GTM model; outliers are characterized by its large values. As an example, the ten largest values are labeled.

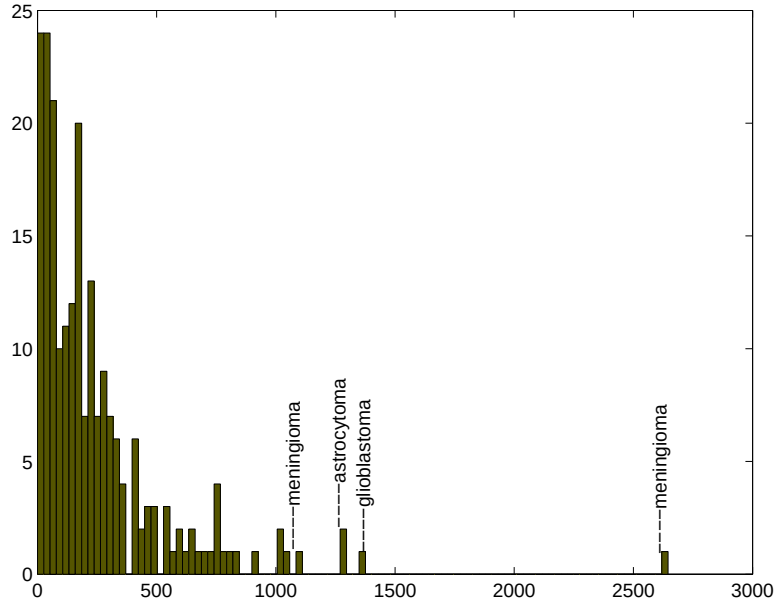

Figure 5.9: Histogram of Eq. 5.9 for class-t-GTM. As an example, the four largest values are labeled.

the proposed fixed initialization strategies across the whole range of possible final number of clusters, while being far more costly in computational terms.

The entropy measure in Eq. 5.7 quantifies the level of agreement between the clustering solutions and the class distributions. In terms of the overall clusterwise class separation provided by the Gaussian distributions-based GTM clustering models, it has been shown that the addition of class information consistently helps. As a result, these class-enriched models would be useful in a semi-supervised setting in which new undiagnosed tumour cases were added to the database.

5.5 Summary

In this chapter, we have carried out an analysis of the influence exerted by the inclusion of class information in a two-stage clustering procedure. We have also introduced and tested different strategies of initialization for the second stage of this procedure. The first stage is based on the manifold learning class-GTM model. The second stage is based on the well-known K-means algorithm, which was initialized either multiple times randomly, or in a fixed manner making use of the prior knowledge provided by class-GTM in the first stage following a novel procedure based on its Magnification Factors and Cumulative Responsibility. The reported experiments have shown that the two-stage random and fixed initializations yield almost identical results in terms of clusterwise class separation, with the former being computationally more costly. It has also been shown that the two-stage clustering procedures based on standard GTM and class-GTM perform better than the direct K-means clustering of the data in terms of this clusterwise class separation and that the inclusion of class information improves the clusterwise class separation. The existence of atypical data or outliers in the human brain tumours MRS dataset under study, and its influence on the clustering process, have also been explored.

We must note that there is a limitation for the two-stage clustering procedure proposed in this chapter. At all time, we have assumed that the sources generating the available data were unimodal and, therefore, that there was an intrinsic continuity property in the data. This is not always the case, as data can

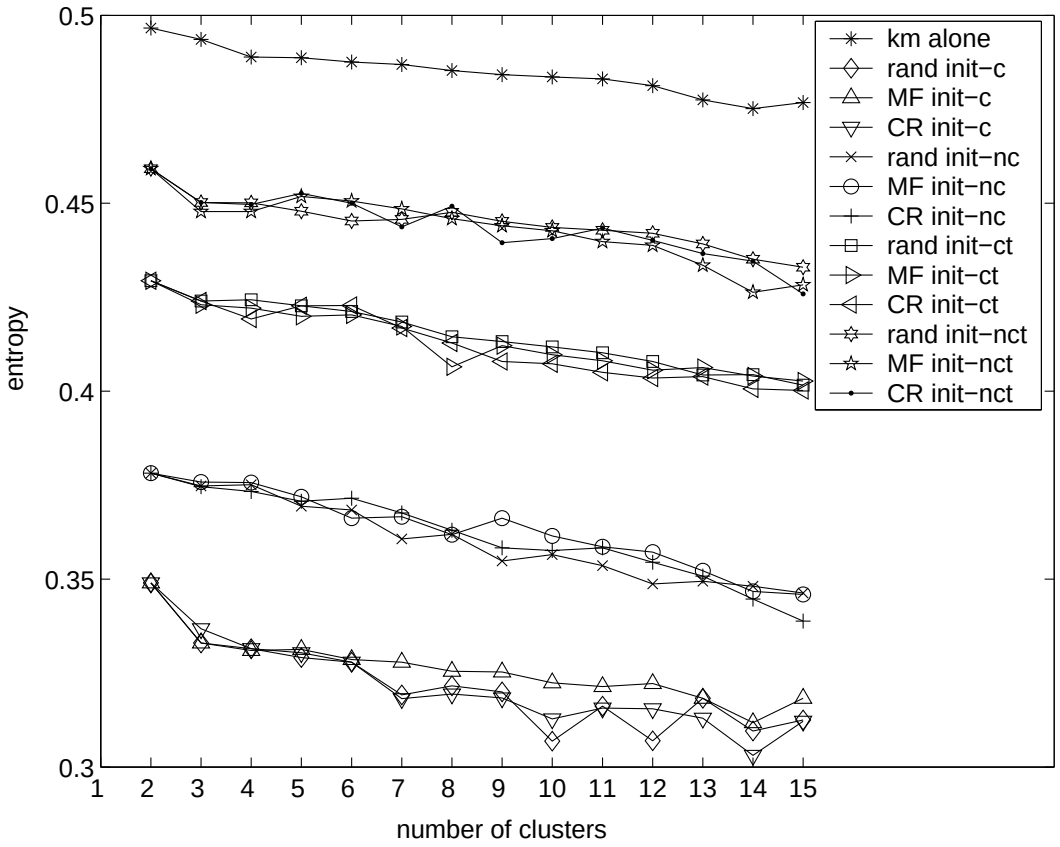

Figure 5.10: Entropies for the clustering of the entire tumour dataset using two-stage clustering with different initializations (based on MF (MF init), CR (CR init) and random (rand init)), and K-means alone. The 'c' symbol means that the corresponding model using class information was used in the first stage and 'nc' for the opposite. The 't' in the legend label means that t-GTM was used in the first stage.

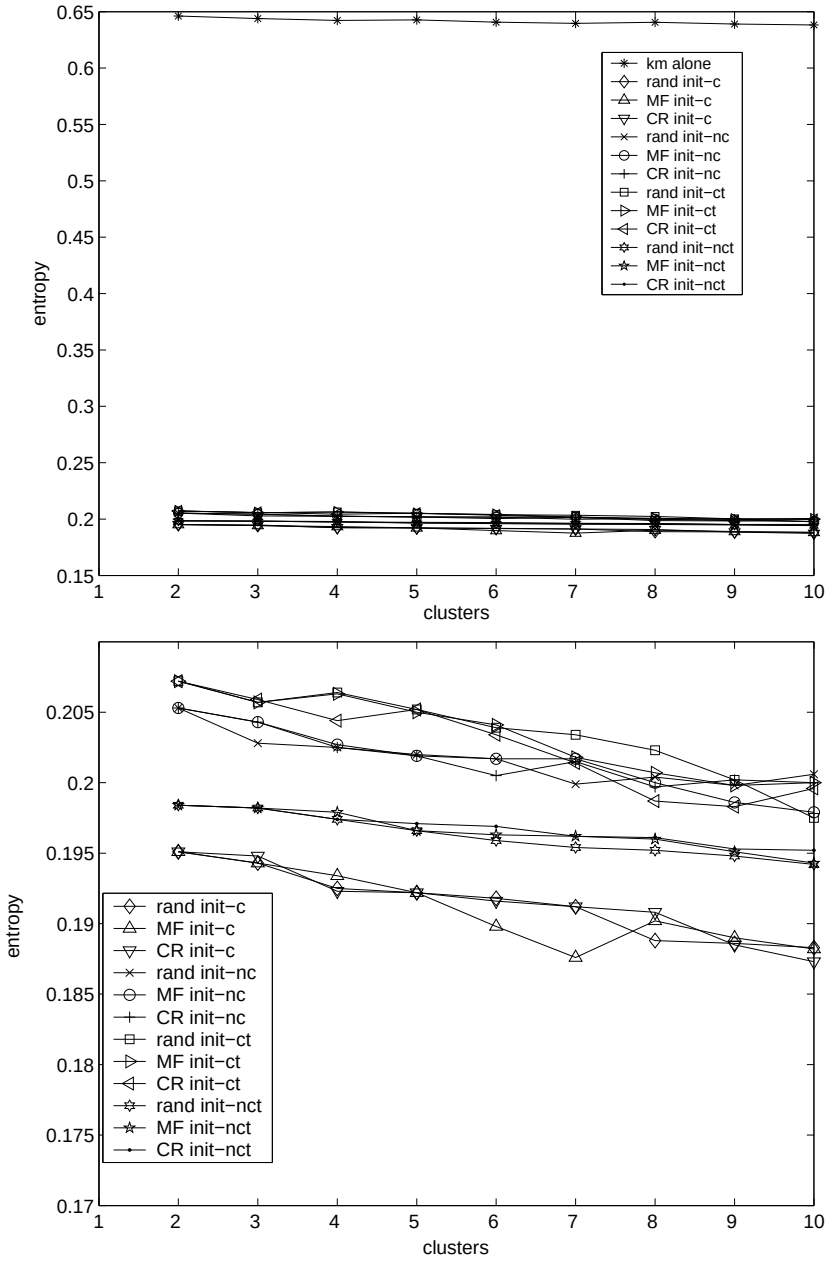

Figure 5.11: Entropy for the two-stage clustering of the second tumour dataset, with different initializations (MF init, CR init and rand init) and K-means alone. The 'c' and 'nc' symbols refer to models that, in turn, use and not use class information. The 't' in the legend means that t-GTM was used in the first stage. (Top): all models are shown. (Bottom): only the GTM, t-GTM and their class-enriched variants are shown.

be at least partially multi-modal. As soon as we introduce class information in the GTM modeling process, data multi-modality, if existing, is likely to generate discordance between the grouping or clustering structure of the data by itself and the distribution of classes. This would be reflected in a worsening of the results in terms of entropy.

Chapter 6

Semi-Supervised Geodesic Generative Topographic Mapping

6.1 Introduction

In many of the databases generated in some of the currently most active areas of research, such as, for instance, biomedicine, bioinformatics, or web mining, class labels are either completely or partially unavailable. The first case scenario is that of unsupervised learning, where the most common task to be performed is that of data clustering, which aims to discover the group structure of multivariate data (Jain & Dubes, 1998). The second case is less frequently considered despite the fact that, quite often, only a reduced number of class labels is readily available and even that can be difficult and/or expensive to obtain. This becomes a task at the interface between supervised and unsupervised models: semi-supervised learning (SSL, Chapelle *et al.* 2006).

As was stated in section 2.1, SSL methods can be developed to assist either classification or clustering tasks mainly. The former task is the purpose of the models described in this chapter, but using a clustering method as a basis.

From the SSL categories summarized in section 2.2, this chapter specifically concerns graph-based methods that use, as a basis, generative unsupervised models for clustering and visualization. As a reminder, in graph-based methods, the nodes of a graph come to represent the observed data points, while its edges are assigned the pairwise distances between the incident nodes. The way the

distance between two data points is computed can be seen as an approximation of the geodesic distance between the two points with respect to the overall data manifold (Belkin & Niyogi, 2004).

In Zhu & Ghahramani (2002), a label propagation (LP) algorithm for SSL was introduced, working under the assumption that close data points tend to have similar class labels. Here, the label of a node (label vector) propagates to neighbouring nodes according to their proximity in a fully connected graph (formed by the input samples, labeled and unlabeled). Thus, labels are propagated through dense unlabeled data regions. An alternative method, (Laplacian Eigenmaps: LapEM) presented in Belkin & Niyogi (2003b), assumes that the data lie on a manifold in a high dimensional space. The learning of the underlying manifolds is accomplished using all the available input samples. A proximity graph is then constructed, using node adjacencies, as a model for the manifold. The proposed graph Laplacian approximates the Laplace-Beltrami operator, which can be thought of as an operator on functions defined on nodes of the proximity graph. Recently, in Herrmann & Ultsch (2007), a two-stage SSL method was proposed. In the first stage, data points are clustered using the Emergent Self-Organizing Map (ESOM, Ultsch 2003). Then, ESOM is considered as a proximity graph and a modified LP is carried out in the second stage.

In this thesis, we present a semi-supervised approach, inspired by that proposed in Herrmann & Ultsch (2007). It is based on Geo-GTM (Cruz-Barbosa & Vellido, 2008d), which is an extension of the statistically principled GTM (Bishop *et al.*, 1998) that has been described in chapter 4 of the thesis. Geo-GTM prioritizes neighbourhood relationships along a generated manifold embedded in the observed data space. This model has been shown to improve both the trustworthiness and the continuity of the low-dimensional data representations, and also to behave robustly in the presence of noise (Cruz-Barbosa & Vellido, 2008a,b). In our proposal, the prototypes are inserted and linked by the nearest neighbour to the data manifold constructed by Geo-GTM. The resulting graph is considered as a proximity graph for which an *ad hoc* version of LP is defined. The resulting semi-supervised Geo-GTM (SS-Geo-GTM) uses the information derived from Geo-GTM training to accomplish the semi-supervised task. Following the same methodology, we have also developed in this thesis a semi-supervised version for

the standard GTM (SS-GTM) and compared its performance with that of SS-Geo-GTM. Several experiments using artificial and real datasets show that the performance of SS-Geo-GTM, measured as a fraction of correctly classified input samples, is significantly better than that of SS-GTM for data sets of convoluted geometry. Also, we compare the performance of the proposed SS-Geo-GTM with that of Laplacian Eigenmaps (a popular graph-based semi-supervised method). Several experiments with artificial and real data sets, using different percentages of available class labels and also with the presence of different levels of uninformative noise, show that SS-Geo-GTM overall outperforms both SS-GTM and LapEM.

In this thesis, we conclude the evaluation of the capabilities of the SS-Geo-GTM model with the analysis of a real and considerably difficult problem: that of inferring survival stages in an aggressive human brain tumour pathology from a very limited amount of available survival stage labels and Magnetic Resonance Spectroscopy (MRS) data corresponding to these tumours. This pathology, namely Glioblastomas, is known for its heterogeneity. To the best of the author knowledge, this approach to survival stage analysis has never been attempted before using this type of data. Here, the performance criterion is the retrieval accuracy of labels, defined as the ratio of correctly retrieved labels to the total number of retrievable labels. The performance of SS-Geo-GTM for the prognostic problem at hand compares favourably with those of SS-GTM and the alternative LapEM models.

6.2 Semi-Supervised Geo-GTM

If only unlabeled data were available and our analyses only concerned data clustering, the previously described Geo-GTM would suffice. In many real situations, though, we may well count with only a limited amount of labeled cases. If this is the case, and we are also interested in classification, the problem can be addressed as a semi-supervised one. The goal in such problem is inferring the unavailable class labels using the information provided by the few available ones as well as by the cluster structure defined by Geo-GTM. The latter is contained in the pro-

totypes $\mathbf{y}_m$, the responsibilities z_{mn}^{geo} defined in Eq. 4.2, and the data manifold obtained for computing the graph distance.

The basic idea underlying the proposed semi-supervised approach is that neighbouring points are most likely to share their label and that these labels are best propagated through neighbouring nodes according to proximity. Assuming that the Geo-GTM prototypes and the corresponding constructed data manifold can be seen as a proximity graph, we modify an existing label propagation (LP) algorithm (Zhu & Ghahramani, 2002) to account for the information provided by the trained Geo-GTM. The result is the proposed semi-supervised Geo-GTM (SS-Geo-GTM, for short).

The LP method is adapted to Geo-GTM as follows. A label vector $\mathbf{L}_m \in [0,1]^k$ is first associated to each Geo-GTM prototype $\mathbf{y}_m$. These label vectors can be considered as nodes in a proximity graph. The weights of the edges are derived from the graph distances d_g between prototypes. For this, the prototypes are inserted and linked to the graph through the nearest data point. It is important to note that, in this process, empty clusters (that is, those associated to a given prototype $\mathbf{y}_m$, to which no data point is assigned, or, in other words, those that do not bear a maximum of responsibility z_{mn}^{geo} for any data point n) are omitted. The edge weight between nodes m and m' is calculated as

$$w_{mm'} = \exp(-\frac{d_g^2(m,m')}{\sigma^2}), \tag{6.1}$$

where the σ parameter defines the level of sparseness in the graph for label information. One possible choice for the value of this parameter is the minimal inter-prototype distance. An alternative choice is defined in section 6.2.1 and evaluated in section 6.3.

Following Herrmann & Ultsch (2007), the available label information of $\mathbf{x}_n \in X$ with class attribution $c(\mathbf{x}_n) = C_t \in \{C_1, \ldots, C_k\}$ will be used to fix the label vectors of the prototypes to which they are assigned ($\mathbf{x}_n$ is assigned to $\mathbf{y}_m$ through $\mathbf{u}_m = \arg\max_{\mathbf{u}_i} z_{in}^{geo}$), so that $L_{m,j} = 1$ if $j = t$, and $L_{m,j} = 0$ otherwise. Unlabeled prototypes will then update their label by propagation according to

$$\mathbf{L}_m^{new} = \frac{\sum_{m'} w_{mm'} \mathbf{L}_{m'}}{\sum_{m'} w_{mm'}}, \tag{6.2}$$

until no further changes occur in the label updating. Subsequently, unlabeled data items are labeled by assignment to the class more represented on the label vector of the prototype $\mathbf{y}_m$ bearing the highest responsibility for them, according to $c(\mathbf{x}_n) = \arg\max_{C_j \in \{C_1,\ldots,C_k\}} L_{m,j}$. The same methodology is used to build a semi-supervised version of a standard GTM model (SS-GTM).

For illustration, the process of computing the graph distances between prototypes is shown in Fig. 6.1 (bottom), using the *Dalí* set described in section 6.3.1.

6.2.1 Choice of the σ Parameter

As stated in Zhu & Ghahramani (2002), an essential problem in LP for semi-supervised learning is finding an adequate value for parameter σ in Eq. 6.1. It is known that for $\sigma \to \infty$, all unclassified data cases are assigned the same label vector because of label vectors shrinking to a single point (with large σ, unlabeled cases tend to have similar class probabilities, then receiving the same influence from all labeled cases). On the other hand, when $\sigma \to 0$, the performance of LP is similar to that of a 1-nearest neighbour classifier. Therefore, a suitable value for the parameter should lie between these two extremes.

Here, we propose an *ad hoc* criterion that consists on assigning σ the value of what we call the main reference inter-prototype (MRIP) distance. For this, we first calculate the Cumulative Responsibility (CR), which is the sum of responsibilities over all data items in X, for each cluster m, $CR_m = \sum_{n=1}^{N} z_{mn}^{geo}$. The prototypes with highest CR are considered as the most representative in the dataset (this was evaluated and showed in chapter 5 and in Cruz-Barbosa & Vellido 2007c,d). We then choose MRIP to be the graph distance $d_g(\mathbf{y}_{m1}, \mathbf{y}_{m2})$ between the two non-contiguous prototypes $\mathbf{y}_{m1}, \mathbf{y}_{m2}$ of highest CR. Note that the use of the graph distance assures the minimal inter-prototype path.

6.2.2 Summary of the SS-Geo-GTM algorithm

For the sake of clarity, we provide in this section some details of the proposed SS-Geo-GTM algorithm. It is assumed that the analysed dataset has previously been modeled by Geo-GTM (as defined en chapter 4) and that the corresponding

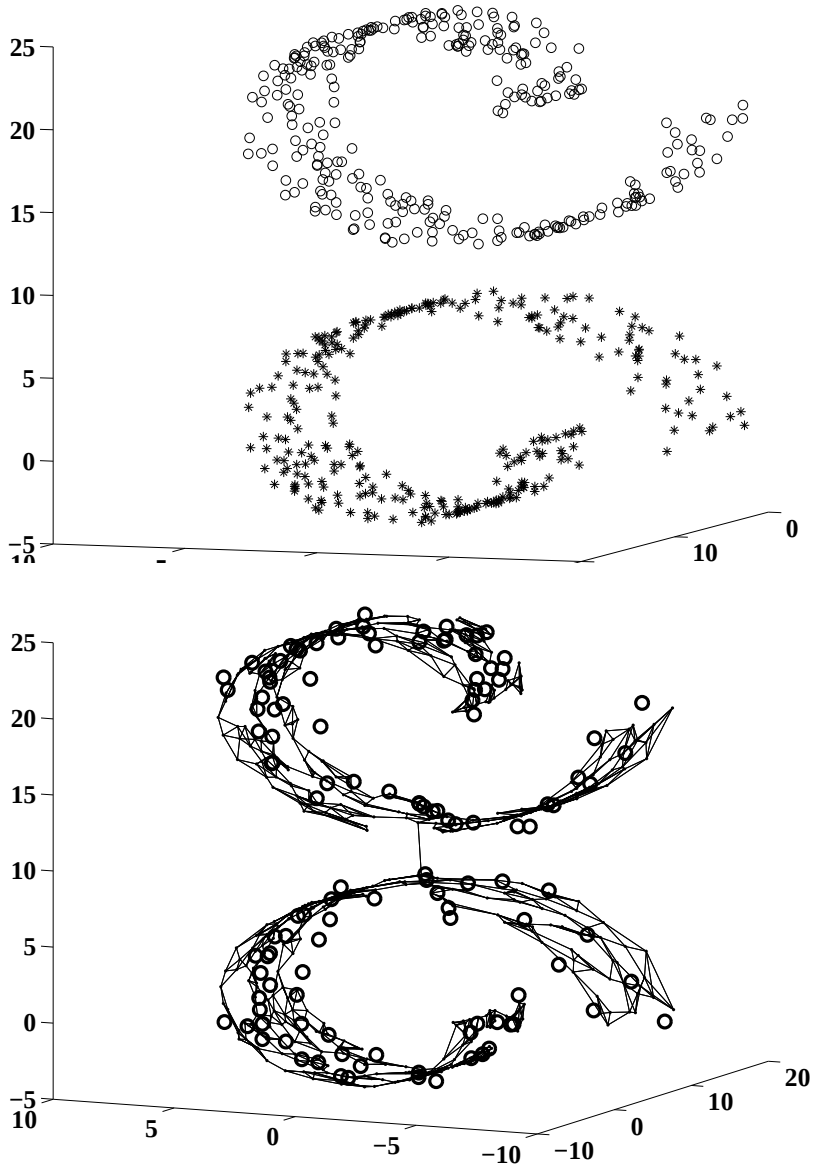

Figure 6.1: (Top): The artificial 3-D *Dalí* dataset, where the two contiguous fragments are assumed to correspond to different classes, identified with different symbols. (Bottom): Results of the Geo-GTM modeling of *Dalí*. The prototypes are represented by '○' symbols (only the non-empty prototypes are preserved and linked to the graph through the nearest data point). The graph constructed using 4-nearest neighbours is represented by lines connecting the data points, which are, in turn, represented by '·' symbols.

cluster structure is provided. This cluster structure contains the M prototypes, the responsibilities z_{mn}^{geo} (defined in Eq. 4.2) and the data manifold obtained for computing the graph distance. Also, the general settings for LP are assumed: the class label availability of a dataset $\mathbf{X}$ (l labeled and u unlabeled data points) is known, and it is assumed that the number of classes C is known and that all classes are present in the labeled data. The LP algorithm, summarized in appendix A.2, has been modified in the following way:

- Pre-processing stage
  - Create a connected graph by inserting and linking the M prototypes to the nearest neighbour of the data manifold constructed by Geo-GTM. Here, the nodes are all prototypes.
  - Compute the graph distance among prototypes using the constructed graph in step 1.
  - Compute the weights w_{ij} of the edges between nodes i, j as in Eq. 6.1, where σ is obtained as showed in section 6.2.1.
  - Compute a $M \times M$ transition matrix T as $T_{ij} = \frac{w_{ij}}{\sum_{k=1}^{l+u} w_{kj}}$, where T_{ij} is the probability of propagation from node j to node i.
  - Define a $(l+u) \times C$ label matrix L, whose ith row represents the label probability distribution of data point $\mathbf{x}_i$.
  - Define a $M \times C$ prototypes label matrix L', whose ith row represents the label probability distribution of node (prototype) $\mathbf{y}_i$. Here, the available label information of $\mathbf{x}_n \in \mathbf{X}$ (given by L) with class attribution $c(\mathbf{x}_n) = C_t \in \{C_1, \ldots, C_k\}$ is used to fix the label vectors of the prototypes to which they are assigned ($\mathbf{x}_n$ is assigned to $\mathbf{y}_m$ through $\mathbf{u}_m = \arg\max_{\mathbf{u}_i} z_{in}^{geo}$), so that $L'_{m,j} = 1$ if $j = t$, and $L'_{m,j} = 0$ otherwise. The initialization of unlabeled nodes is not relevant.

SS-Geo-GTM algorithm

1. Propagate $L' \leftarrow TL'$, as in Eq. 6.2.
2. Row-normalize L' as $L'_{ij} = L'_{ij} / \sum_k L'_{ik}$.

3. Clamp the labeled data. Repeat from step 1 until L' converges.

Finally, unlabeled data points in L are labeled by assignment to the class more represented on the label vector of the prototype $\mathbf{y}_m$ bearing the highest responsibility for them, according to $c(\mathbf{x}_n) = \arg\max_{C_j \in \{C_1,\ldots,C_k\}} L'_{m,j}$.

6.3 Experiments on Standard Datasets

In this section, we first describe the experimental design and settings. This is followed by a presentation and discussion of the corresponding results.

6.3.1 Experimental Design and Settings

Geo-GTM, SS-Geo-GTM, and SS-GTM were implemented in MATLAB®. For the experiments reported next, the adaptive matrix $\mathbf{W}$ was initialized, following a procedure described in Bishop *et al.* (1998), as to minimize the difference between the prototype vectors $\mathbf{y}_m$ and the vectors that would be generated in data space by a partial Principal Component Analysis (PCA). The inverse variance β was initialised to be the inverse of the 3^{rd} PCA eigenvalue. This initialization ensures the replicability of the results. The latent grid was fixed to a square layout of approximately $(N/2)^{1/2} \times (N/2)^{1/2}$, where N is the number of points in the dataset. The corresponding grid of basis functions was equally fixed to a 5×5 square layout for all datasets.

Three datasets were selected for the reported experiments, where two of them (*Dalí* and *Oil-Flow*) can be represented by two-dimensional manifolds:

- The first one is the artificial 3-D *Dalí* set (inspired by one of the common patterns in Salvador Dalí's artworks), as shown in Fig.6.1(top). It consists of two groups of 300 data points each that are images of the functions $\mathbf{x}_1 = (t\cos(t), t_2, t\sin(t))$ and $\mathbf{x}_2 = (t\cos(t), t_2, -t\sin(t) + 20)$, where t and t_2 follow $\mathcal{U}(\pi, 3\pi)$ and $\mathcal{U}(0, 10)$, respectively.

- The second set is the well-known *Iris* data, available from the UCI repository (Asuncion & Newman, 2007), which consists of 150 4-dimensional

items representing several measurements of Iris flowers, which belong to 3 different classes.

- The third is the more complex *Oil-Flow* set, also available online[1], which simulates measurements in an oil pipe corresponding to three possible configurations (classes). It consists of 1,000 items described by 12 attributes.

The central goal of the experiments is the comparison of the performances of SS-Geo-GTM and SS-GTM in terms of classification accuracy. We hypothesize that SS-GTM will yield lower rates of classification accuracy in the semi-supervised task than its geodesic distance-based counterpart, especially for datasets of convoluted geometry such as *Dalí* and *Oil-Flow*.

We first assume that the choice of the MRIP, described in section 6.2.1, as a value for σ is appropriate. In this setting, we evaluate the models in the most extreme semi-supervised setting, that is, when the class label is only available for a single input sample for each class and the remaining samples are considered as unlabeled data. We then proceed to evaluate the performance of the models in this same setting for a range of different values of σ, both higher and lower than the MRIP. With this, it should be possible to assess the adequacy of the MRIP choice. In the next step, the label availability condition is relaxed, and the models are evaluated in the presence of higher ratios of labels.

Finally, we aim to gauge and compare the robustness of the methods in the presence of noise. In previous research (Cruz-Barbosa & Vellido, 2008a), the Geo-GTM model has been shown to behave better in this respect than the standard GTM model (with the Euclidean metric) as measured by the test log-likelihood. In the semi-supervised extension defined in this thesis, the performance criterion is the classification accuracy.

6.3.2 Results and Discussion

All datasets are first modeled using GTM and Geo-GTM. SS-GTM and SS-Geo-GTM are then built on top of these. As mentioned in the previous section, at first only a single randomly selected input sample per class is kept labeled in each

[1] http://research.microsoft.com/∼cmbishop/PRML/webdatasets/datasets.htm

Dataset	**SS-Geo-GTM** (% ± std)	**SS-GTM** (% ± std)
*Dalí**	99.54± 2.24	90.71± 7.99
*Iris***	88.71 ± 7.88	85.74 ± 8.72
*Oil-Flow**	77.43±8.31	36.74±3.29

Table 6.1: Classification accuracy as an average percentage over hundred runs (with its corresponding standard deviation). The semi-supervised models are here presented with a single randomly selected labeled input sample per class. The statistical significance of the differences between SS-Geo-GTM and SS-GTM is indicated with '*' if $p < 0.01$ and with '**' if $p < 0.05$.

run of the algorithm, while the remaining samples are considered as unlabeled data. The semi-supervised performance of the models is measured as the average percentage of correctly classified input samples over one hundred runs. The corresponding results are shown in Table 6.1. SS-Geo-GTM significantly (according to a one-way analysis of variance ANOVA) outperforms SS-GTM in all datasets and, as hypothesized, most notoriously for the datasets of more convoluted geometry, namely *Dalí* and *Oil-Flow*. The differences are less notorious for the less convoluted *Iris* dataset. The extreme differences observed for the *Oil-Flow* set are of special interest, given that its high dimensionality precludes straightforward exploratory visualization.

As stated in section 6.3.1, the previous results are obtained by setting the value of $\sigma = MRIP$. We then proceed to evaluate the performance of the models in this same setting for a range of different values of σ, both higher and lower than the MRIP, to assess the adequacy of the proposed MRIP choice. We explore the interval $\sigma \in [MRIP - \epsilon, MRIP + \epsilon]$, where $\epsilon > 0$, and measure the performance of SS-Geo-GTM over a hundred runs. These results are reported in Table 6.2. The models with $\sigma = MRIP$ yield the best results in the range of selected σ values, which confirms the fact that the MRIP value is at least near the optimum value for σ. Consequently, from here on MRIP will be used as the default value for σ.

The proposed SS-Geo-GTM model has been shown to perform well and better than the SS-GTM in the most extreme semi-supervised setting. The question

Dalí		*Iris*		*Oil-Flow*	
$\sigma < MRIP$	% ± std	$\sigma < MRIP$	% ± std	$\sigma < MRIP$	% ± std
5.0	98.06±3.73	0.05	85.72±8.93	0.10	74.74±8.63
10.0	98.46±4.69	0.10	87.24±8.97	0.20	75.03±9.08
15.0	99.19±2.44	0.12	87.37±7.46	0.25	75.24±9.26
20.0	99.37±2.22	0.14	86.94±9.73	0.30	74.38±10.10
25.0	99.48±2.13	0.15	88.20±8.14	0.35	75.74±8.98
MRIP = 31.36	99.54±2.24	**MRIP**= 0.21	88.71±7.88	**MRIP** = 0.43	77.43±8.31
$\sigma > MRIP$		$\sigma > MRIP$		$\sigma > MRIP$	
35.0	98.54±3.96	0.30	88.30±7.46	0.50	75.97±8.51
40.0	98.43±4.54	0.40	88.69±8.93	0.55	74.71±8.56
45.0	97.95±4.77	1.0	88.64±7.63	0.60	74.70±8.80
50.0	96.84±6.55	3.0	88.59±5.32	0.65	73.98±8.77
55.0	95.35±8.01	4.0	83.03±7.29	0.75	72.08±9.88

Table 6.2: Average classification accuracy (as a percentage) and its standard deviation over one hundred runs for different values of σ parameter in the SS-Geo-GTM setting.

remains: will this difference of performance remain the same when the label availability condition is relaxed? To answer this question, the ratio of randomly selected labeled data is increased from a single one to a 1%, and from there, up to a 10%. The experiment is again carried out a hundred times for each dataset. The corresponding results are shown in Table 6.3.

SS-Geo-GTM clearly (and again significantly according to an ANOVA test) outperforms SS-GTM for *Dalí* and *Oil-Flow* and, as expected, the performance monotonically improves with the increasing percentage of labels. The differences for the latter set, more complex and high-dimensional, are striking. For *Dalí*, SS-Geo-GTM achieves a 100% accuracy even with a 1% of labeled data, while SS-GTM does not reach that average accuracy even with a 10%. The *Iris* data set benefits less of the addition of class labels and the performances of both models are comparable. This is consistent with the previous results and supports the hypothesis that the use of the geodesic metric is likely to improve the results mainly for data sets of convoluted underlying geometry. Notice also that the standard deviation from the mean results monotonically decreases for all datasets with the increasing percentage of available labels, reducing the uncertainty of the

% of avail. labels	Classification accuracy (% ± std)					
	*Dalí**		*Iris***		*Oil-Flow**	
	SS-Geo	SS-GTM	SS-Geo	SS-GTM	SS-Geo	SS-GTM
1	100±0	93.43±5.46	⋆	⋆	83.93±5.60	39.96±3.44
2	100±0	96.96±3.41	⋆	⋆	90.08±3.49	55.88±10.95
3	100±0	97.74±2.05	⋆	⋆	91.79±3.07	64.71±7.95
4	100±0	98.29±1.80	90.00±8.11	89.46±5.24	94.28±2.60	70.69±6.06
5	100±0	98.61±1.32	89.96±6.98	89.18±6.48	95.14±2.20	74.11±5.05
6	100±0	98.66±1.64	91.30±7.37	91.66±3.02	95.97±2.01	76.51±4.30
7	100±0	98.98±0.80	90.74±7.62	90.94±3.03	96.43±1.81	79.10±4.24
8	100±0	99.19±0.82	91.91±5.31	91.90±3.03	96.65±1.53	80.88±4.27
9	100±0	99.30±0.70	92.35±4.90	91.88±2.48	97.11±1.66	82.19±3.43
10	100±0	99.24±0.73	93.19±4.36	92.32±2.42	97.53±1.22	83.91±3.58

Table 6.3: Average classification accuracy (as a percentage) and its standard deviation over one hundred runs, for SS-GeoGTM and SS-GTM. A randomly increasing percentage of pre-labeled items per class was chosen in each run. The '⋆' symbol replacing results means that the experiment was not carried out because the corresponding percentage of available labels was less than or equal to one label per class. A super-index '*' indicates that the differences between both models were significant at level $p < 0.01$ in the ANOVA test for all percentages of class label availability. A super-index '**' indicate that no differences were significant.

results.

It was shown in Cruz-Barbosa & Vellido (2008a) that Geo-GTM can recover the true underlying data structure far better than the standard GTM (as reflected in a lower test log-likelihood), even in the presence of a considerable amount of noise in the data. We now extend these results to the semi-supervised setting to gauge and compare the robustness of the analyzed methods in the presence of noise in some illustrative experiments. For this, Gaussian noise of zero mean and increasing standard deviation was added to: a noise-free version of the *Dalí* set (added noise from $\sigma = 0.1$ to $\sigma = 2.0$, partially illustrated in left column of Fig. 6.2) and the most difficult dataset, *Oil-Flow* (added noise from $\sigma = 0.01$ to $\sigma = 0.2$, partially illustrated in right column of Fig. 6.2). As in the previous

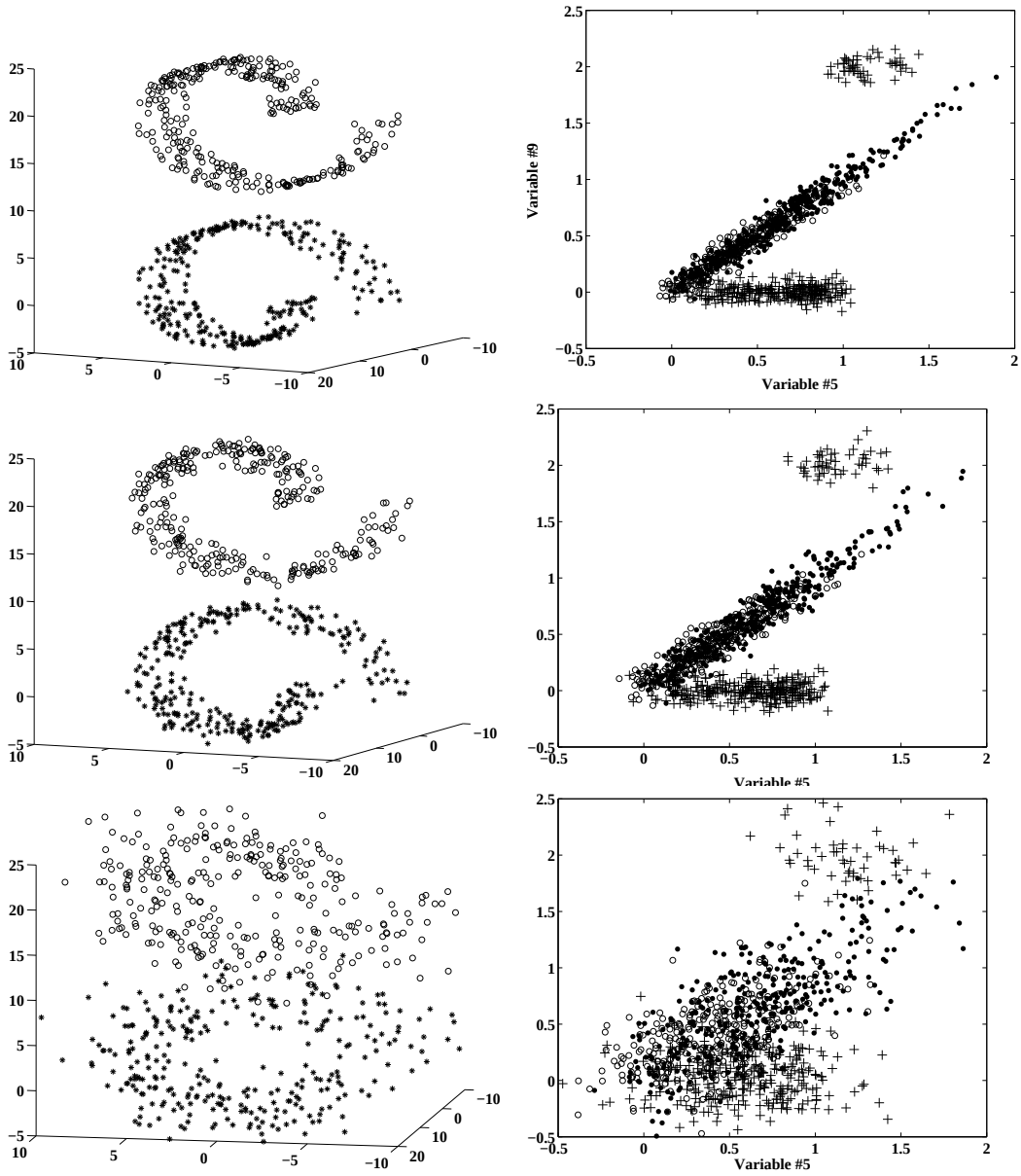

Figure 6.2: Noisy variations of some of the data used in the experiments, provided for illustration. The noise scale magnitude is in correspondence with the data scale. For *Dalí*, from top-left to bottom-left, noise of standard deviations $\sigma = 0.1$, $\sigma = 0.5$, and $\sigma = 2.0$. For *Oil-Flow*, we provide three views of variable 5 versus variable 9: From top-right to bottom-right, noise of standard deviations $\sigma = 0.01$, $\sigma = 0.05$, and $\sigma = 0.2$.

Dataset	noise level	model	Percent of available labels				
			2	**4**	**6**	**8**	**10**
Dalí	0.1	SS-Geo	100±0	100±0	100±0	100±0	100±0
		SS-GTM	96.29±3.37	98.15±1.97	99.09±1.0	99.31±0.99	99.28±0.89
	0.3	SS-Geo	99.83±1.11	100±0	100±0	100±0	100±0
		SS-GTM	95.57±4.0	98.11±1.45	98.56±0.83	98.77±0.75	98.88±0.69
	0.5	SS-Geo	99.04±3.16	100±0	100±0	100±0	100±0
		SS-GTM	96.52±3.09	98.05±2.16	98.99±1.40	99.31±1.06	99.39±0.78
	1.0	SS-Geo	95.14±5.52	97.75±2.94	98.71±1.98	99.23±0.73	99.28±0.92
		SS-GTM	96.12±3.79	98.36±1.53	98.66±1.21	99.04±0.45	99.06±0.35
	2.0	SS-Geo	94.78±3.66	96.45±1.63	96.96±0.67	97.11±0.58	97.19±0.48
		SS-GTM	92.96±3.0	94.28±1.96	94.73±1.75	95.45±1.01	95.36±1.07
Oil-Flow	0.01	SS-Geo	88.13±4.05	93.87±2.71	95.63±2.24	96.87±1.45	97.26±1.18
		SS-GTM	55.54±11.94	70.66±5.84	77.14±4.65	80.25±3.58	84.15±3.39
	0.03	SS-Geo	88.60±4.06	93.34±2.94	95.46±1.94	96.31±1.64	96.98±1.23
		SS-GTM	55.14±10.71	71.54±6.00	77.26±4.53	81.40±3.63	82.60±3.24
	0.05	SS-Geo	90.10±4.38	94.94±2.49	96.34±1.93	97.42±1.69	97.84±1.23
		SS-GTM	53.39±11.81	70.52±7.42	75.79±4.77	81.32±4.52	83.84±4.34
	0.1	SS-Geo	60.40±12.81	81.48±8.91	88.95±4.89	91.19±3.59	92.49±2.59
		SS-GTM	49.88±10.11	70.30±8.63	78.20±4.48	82.68±4.50	85.08±4.23
	0.2	SS-Geo	59.89±11.38	75.76±6.16	79.50±5.03	83.0±3.78	85.41±2.63
		SS-GTM	44.94±9.92	56.18±10.59	66.01±7.04	72.31±5.55	75.37±4.27

Table 6.4: Average classification accuracy (as a percentage) and its standard deviation over one hundred runs, for SS-GeoGTM and SS-GTM models in the presence of increasing levels of uninformative noise. An increasing percentage of pre-labeled items per class was randomly chosen in each run.

experiment, we also analyze the evolution of the performance of these models as the percentage of available labels for each dataset is increased from 2% to 10%.

These new results are shown in Table 6.4. In accordance to the results presented in Cruz-Barbosa & Vellido (2008a), the geodesic variant SS-Geo-GTM consistently outperforms SS-GTM across data sets and noise levels (with a couple of exceptions for high noise levels in *Dalí*). The robustness of the semi-supervised procedure for SS-GTM is surprisingly good, though. This means that, even if SS-GTM is worst at recovering the underlying data structure, the label propagation procedure is affected by noise in a similar manner for both models. For the more complex *Oil-Flow* set, both models deteriorate significantly at high noise levels. Overall, these results seem to indicate that the resilience of the models is mostly due to the inclusion of the geodesic metric and not to the semi-supervised procedure itself.

6.4 Experimental comparison of SS-Geo-GTM with Laplacian Eigenmaps

In this section, a comparison of the previous results of SS-Geo-GTM and SS-GTM with the alternative Laplacian Eigenmaps (LapEM) method is presented. First, the LapEM method is described. Then, the corresponding results of the comparison are shown.

6.4.1 Laplacian Eigenmaps

Laplacian Eigenmaps (LapEM) were defined under the assumption that the observed data lie on a low-dimensional submanifold of the high-dimensional data space (Belkin & Niyogi, 2003b). As a model for a manifold (in the sense of Riemannian manifolds), an adjacency graph is constructed using the input data points as nodes. Edge weights between nodes i, j can be derived from the distances between the corresponding nodes or simply by taking $w_{i,j} = 1$ if data points x_i and x_j are connected, and $w_{ij} = 0$ otherwise. Then, in order to exploit the structure of the model, the graph Laplacian L is obtained for the adjacency graph. L is a symmetric, positive semidefinite matrix which can be thought of as

an operator (Laplace Beltrami operator) on functions defined on vertices of the graph.

The classifier is constructed using the eigenfunctions of the Laplace Beltrami operator, which provide a natural basis for functions on the manifold. In other words, only input data points (labeled and unlabeled) information is needed to recover the manifold. Then, the labeled data are used to develop a classifier defined on this manifold.

6.4.2 Results and Discussion

Geo-GTM, SS-Geo-GTM, and SS-GTM were initialized following a procedure described in Bishop *et al.* (1998). The latent grid was fixed to a square layout of approximately $(N/2)^{1/2} \times (N/2)^{1/2}$, where N is the number of points in the data set.

The same three data sets described in section 6.3.1 were selected for the reported experiments: *Dalí*, *Iris* and *Oil-Flow*.

The central goal of the experiments is the comparison of the performances of SS-Geo-GTM, SS-GTM and the alternative method of Laplacian Eigenmaps (LapEM, Belkin & Niyogi 2003b) in terms of classification accuracy. We then evaluate (average accuracy over one hundred runs) the models in the most extreme semi-supervised setting: when the class label is available for only one input item for each class while the rest is unlabeled. The corresponding results are shown in Table 6.5. SS-Geo-GTM significantly outperforms SS-GTM and LapEM for all data sets and, most notoriously, for the data sets of more convoluted geometry. The differences with SS-GTM are less notorious for the less convoluted *Iris* data set. LapEM yields a very poor behaviour in this setting.

As in section 6.3.2, the label availability condition is relaxed in order to assess whether the difference of performance remain. Then, the ratio of randomly selected labeled data is increased from a single one to a 1%, and from there, up to a 10%. The corresponding results are shown in Table 6.6. SS-Geo-GTM clearly (and again significantly according to an ANOVA test) outperforms SS-GTM for *Dalí* and *Oil-Flow* and, as expected, the performance monotonically improves with the increasing percentage of labels. The differences for the latter set, more

data set	**SS-Geo-GTM** (% ± std)	**SS-GTM** (% ± std)	**LapEM** (% ± std)
*Dalí**	99.54± 2.24	90.71± 7.99	54.57±3.13
*Iris***	88.71 ± 7.88	85.74 ± 8.72	50.39±3.37
*Oil-Flow**	77.43±8.31	36.74±3.29	63.50±12.08

Table 6.5: Classification accuracy as an average percentage over one hundred runs (with its corresponding standard deviation). The statistical significance (calculated through a one-way ANOVA test) of the differences between SS-Geo-GTM and SS-GTM is indicated with '*' if $p < 0.01$ and with '**' if $p < 0.05$. Also, $p < 0.01$ was obtained between any SS version and LapEM.

complex and high-dimensional, are striking. Also, SS-Geo-GTM outperforms LapEM for all data sets. For *Dalí*, SS-Geo-GTM achieves a 100% accuracy even with a 1% of labeled data, while SS-GTM and LapEM do not reach that average accuracy even with a 10%. The *Iris* data set benefits less of the addition of class labels and the performances of SS-Geo-GTM and SS-GTM models are comparable. This confirms that the use of the geodesic metric is likely to improve the results mainly for data sets of convoluted underlying geometry.

We now extend these results, as in Cruz-Barbosa & Vellido (2008a) and section 6.3.2, to the semi-supervised setting to gauge and compare the robustness of the analyzed methods in the presence of noise in some illustrative experiments. For this, Gaussian noise of zero mean and increasing standard deviation was added to: a noise-free version of the *Dalí* set (added noise from $\sigma = 0.1$ to $\sigma = 2.0$) and the most difficult dataset, *Oil-Flow* (added noise from $\sigma = 0.01$ to $\sigma = 0.2$). The noise scale magnitude is in correspondence with the data scale. As in the previous experiment, we also analyze the evolution of the performance of these models as the percentage of available labels for each dataset is increased from 2% to 10%.

These new results are shown in Table 6.7. In accordance to the results presented in Cruz-Barbosa & Vellido (2008a), the geodesic variant SS-Geo-GTM consistently outperforms SS-GTM (and LapEM) across data sets and noise levels, with few exceptions. The robustness of the semi-supervised procedure for SS-GTM is surprisingly good, though. For the more complex *Oil-Flow* set, both models deteriorate significantly at high noise levels. Overall, these results indi-

% of avail. labels	**Classification accuracy** (% ± std)					
	Dalí			*Iris*		
	SS-Geo	SS-GTM *	LapEM *	SS-Geo	SS-GTM **	LapEM *
1	100±0	93.43±5.46	64.91±4.52	⋆	⋆	⋆
2	100±0	96.96±3.41	76.00±5.88	⋆	⋆	⋆
3	100±0	97.74±2.05	79.65±9.29	⋆	⋆	⋆
4	100±0	98.29±1.80	75.24±10.56	90.00±8.11	89.46±5.24	58.10±4.01
5	100±0	98.61±1.32	88.72±8.05	89.96±6.98	89.18±6.48	57.01±4.57
6	100±0	98.66±1.64	95.01±4.95	91.30±7.37	91.66±3.02	63.68±4.48
7	100±0	98.98±0.80	97.68±3.16	90.74±7.62	90.94±3.03	64.22±4.86
8	100±0	99.19±0.82	98.64±2.13	91.91±5.31	91.90±3.03	69.84±5.26
9	100±0	99.30±0.70	98.88±1.87	92.35±4.90	91.88±2.48	70.19±4.97
10	100±0	99.24±0.73	99.39±1.39	93.19±4.36	92.32±2.42	74.87±5.92

% of avail. labels	*Oil-Flow* (% ± std)		
	SS-Geo	SS-GTM *	LapEM *
1	83.93±5.60	39.96±3.44	76.43±7.55
2	90.08±3.49	55.88±10.95	83.36±5.48
3	91.79±3.07	64.71±7.95	87.56±4.42
4	94.28±2.60	70.69±6.06	89.71±3.51
5	95.14±2.20	74.11±5.05	91.63±3.25
6	95.97±2.01	76.51±4.30	92.63±2.76
7	96.43±1.81	79.10±4.24	93.77±2.36
8	96.65±1.53	80.88±4.27	94.41±2.11
9	97.11±1.66	82.19±3.43	95.18±2.07
10	97.53±1.22	83.91±3.58	95.58±1.53

Table 6.6: Average classification accuracy and its std. deviation over 100 runs, for all models. A randomly increasing percentage of pre-labeled items per class was chosen in each run. The '⋆' symbol means that the experiment was not carried out because the corresponding percentage of available labels was less than or equal to one label per class. A super-index '*' indicates that the differences between the corresponding model and SS-Geo-GTM were significant at $p < 0.01$ in the ANOVA test for all percentages of class labels. A super-index '**' indicates that no differences were significant.

Dataset	noise level	model	Percent of available labels				
			2	4	6	8	10
Dalí	0.1	**SS-Geo**	**100±0**	**100±0**	**100±0**	**100±0**	**100±0**
		SS-GTM	96.29±3.37	98.15±1.97	99.09±1.0	99.31±0.99	99.28±0.89
		LapEM	*75.48±6.56*	*75.73±10.38*	*94.48±4.66*	*98.07±2.02*	*98.50±1.96*
	0.3	**SS-Geo**	**99.83±1.11**	**100±0**	**100±0**	**100±0**	**100±0**
		SS-GTM	95.57±4.0	98.11±1.45	98.56±0.83	98.77±0.75	98.88±0.69
		LapEM	*74.47±5.27*	*75.11±11.11*	*95.55±4.82*	*99.03±1.96*	*99.54±1.12*
	0.5	**SS-Geo**	**99.04±3.16**	**100±0**	**100±0**	**100±0**	**100±0**
		SS-GTM	96.52±3.09	98.05±2.16	98.99±1.40	99.31±1.06	99.39±0.78
		LapEM	*77.67±6.79*	*76.56±10.30*	*95.06±4.53*	*97.49±2.76*	*98.87±1.61*
	1.0	**SS-Geo**	**95.14±5.52**	**97.75±2.94**	**98.71±1.98**	**99.23±0.73**	**99.28±0.92**
		SS-GTM	96.12±3.79	98.36±1.53	98.66±1.21	99.04±0.45	99.06±0.35
		LapEM	*73.86±6.07*	*70.73±10.57*	*92.15±5.34*	*97.23±3.09*	*98.93±1.39*
	2.0	**SS-Geo**	**94.78±3.66**	**96.45±1.63**	**96.96±0.67**	**97.11±0.58**	**97.19±0.48**
		SS-GTM	92.96±3.0	94.28±1.96	94.73±1.75	95.45±1.01	95.36±1.07
		LapEM	*74.02±5.72*	*72.11±11.66*	*90.00±5.91*	*94.54±3.37*	*95.99±1.86*
Oil-Flow	0.01	**SS-Geo**	**88.13±4.05**	**93.87±2.71**	**95.63±2.24**	**96.87±1.45**	**97.26±1.18**
		SS-GTM	55.54±11.94	70.66±5.84	77.14±4.65	80.25±3.58	84.15±3.39
		LapEM	*81.35±5.67*	*88.17±3.41*	*91.80±2.67*	*93.20±2.30*	*94.77±1.70*
	0.03	**SS-Geo**	**88.60±4.06**	**93.34±2.94**	**95.46±1.94**	**96.31±1.64**	**96.98±1.23**
		SS-GTM	55.14±10.71	71.54±6.00	77.26±4.53	81.40±3.63	82.60±3.24
		LapEM	*79.79±7.18*	*90.50±3.72*	*94.00±2.72*	*95.91±1.98*	*96.59±1.13*
	0.05	**SS-Geo**	**90.10±4.38**	**94.94±2.49**	**96.34±1.93**	**97.42±1.69**	**97.84±1.23**
		SS-GTM	53.39±11.81	70.52±7.42	75.79±4.77	81.32±4.52	83.84±4.34
		LapEM	*78.26±7.82*	*92.04±2.81*	*94.86±2.22*	*95.79±1.68*	*96.62±1.37*
	0.1	**SS-Geo**	**60.40±12.81**	**81.48±8.91**	**88.95±4.89**	**91.19±3.59**	**92.49±2.59**
		SS-GTM	49.88±10.11	70.30±8.63	78.20±4.48	82.68±4.50	85.08±4.23
		LapEM	*66.78±11.12*	*87.81±4.79*	*92.50±2.95*	*94.23±2.23*	*95.42±1.78*
	0.2	**SS-Geo**	**59.89±11.38**	**75.76±6.16**	**79.50±5.03**	**83.0±3.78**	**85.41±2.63**
		SS-GTM	44.94±9.92	56.18±10.59	66.01±7.04	72.31±5.55	75.37±4.27
		LapEM	*63.75±7.44*	*77.32±4.55*	*82.22±3.31*	*85.47±2.15*	*86.58±1.84*

Table 6.7: Average classification accuracy and its std. deviation over 100 runs, for all models in the presence of increasing levels of uninformative noise. An increasing percentage of pre-labeled items per class was randomly chosen in each run. Bold and italic lettering is used to distinguish between the results of the models and ease their interpretation.

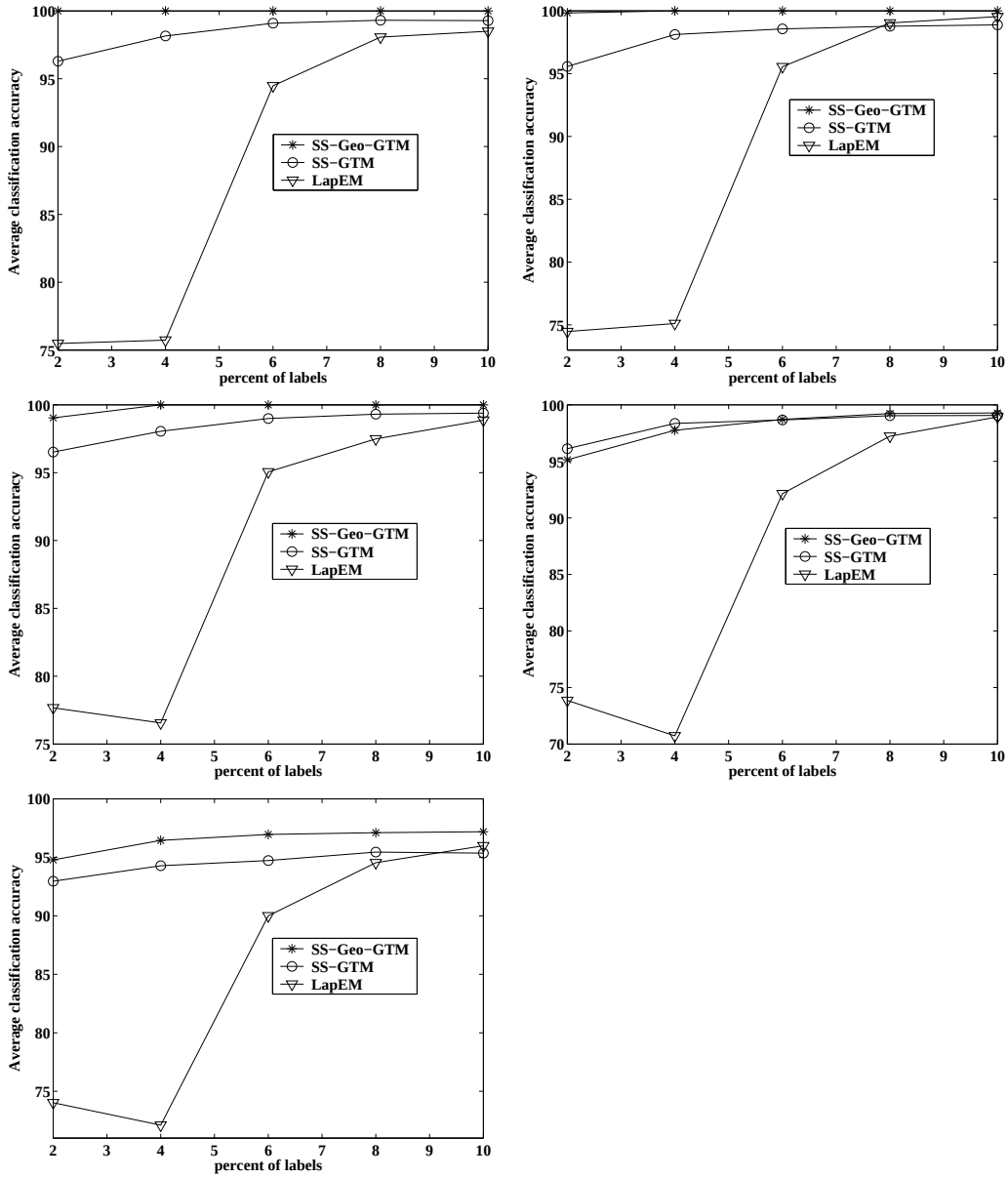

Figure 6.3: Average classification accuracy results taken from Table 6.7 using different and increasing levels of noise for *Dalí* set. From left to right and from top to bottom, noise of standard deviations $\sigma = 0.1, 0.3, 0.5, 1.0, 2.0$.

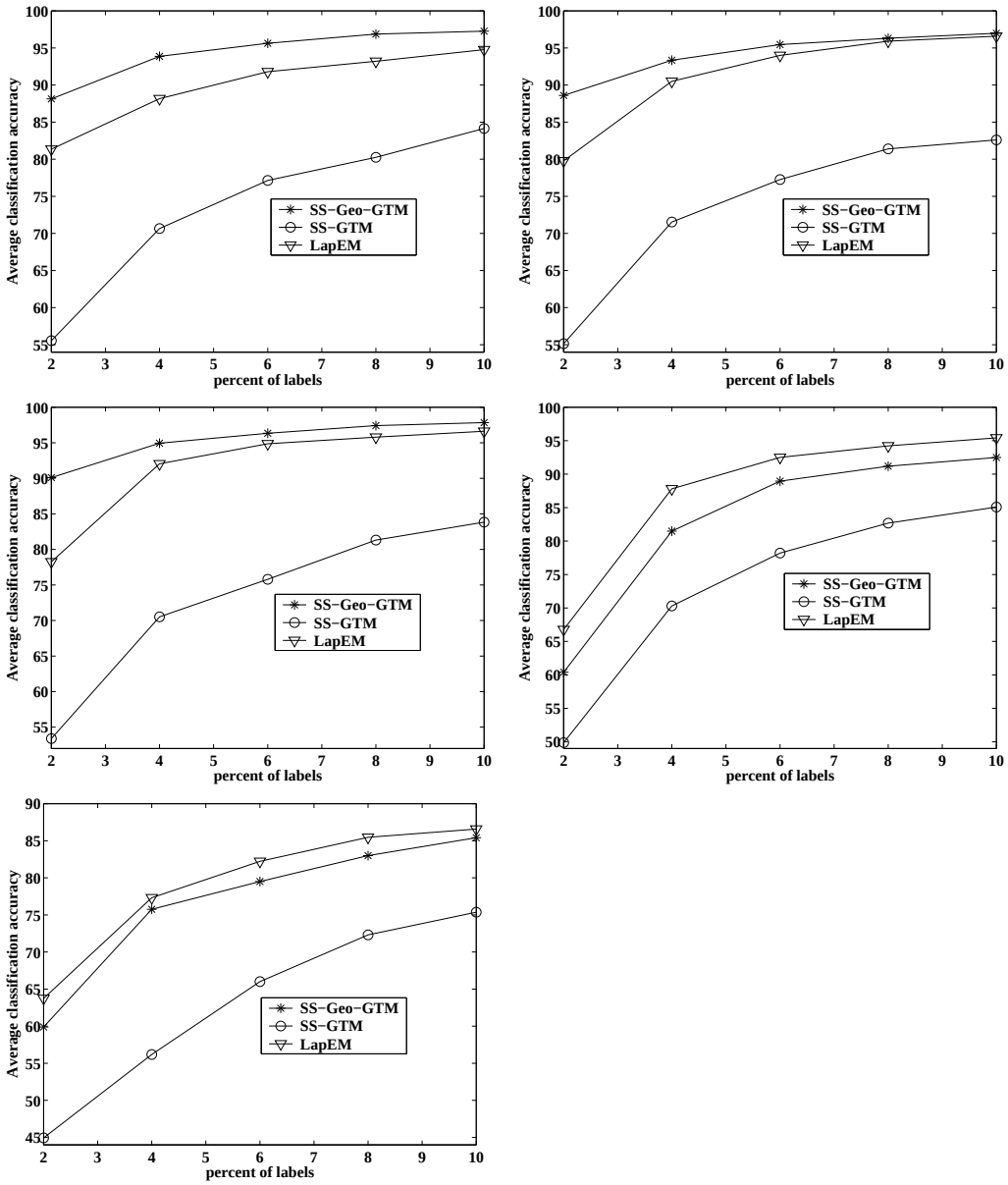

Figure 6.4: Average classification accuracy results taken from Table 6.7 using different and increasing levels of noise for *Oil-Flow* set. From left to right and from top to bottom, noise of standard deviations $\sigma = 0.01, 0.03, 0.05, 0.1, 0.2$.

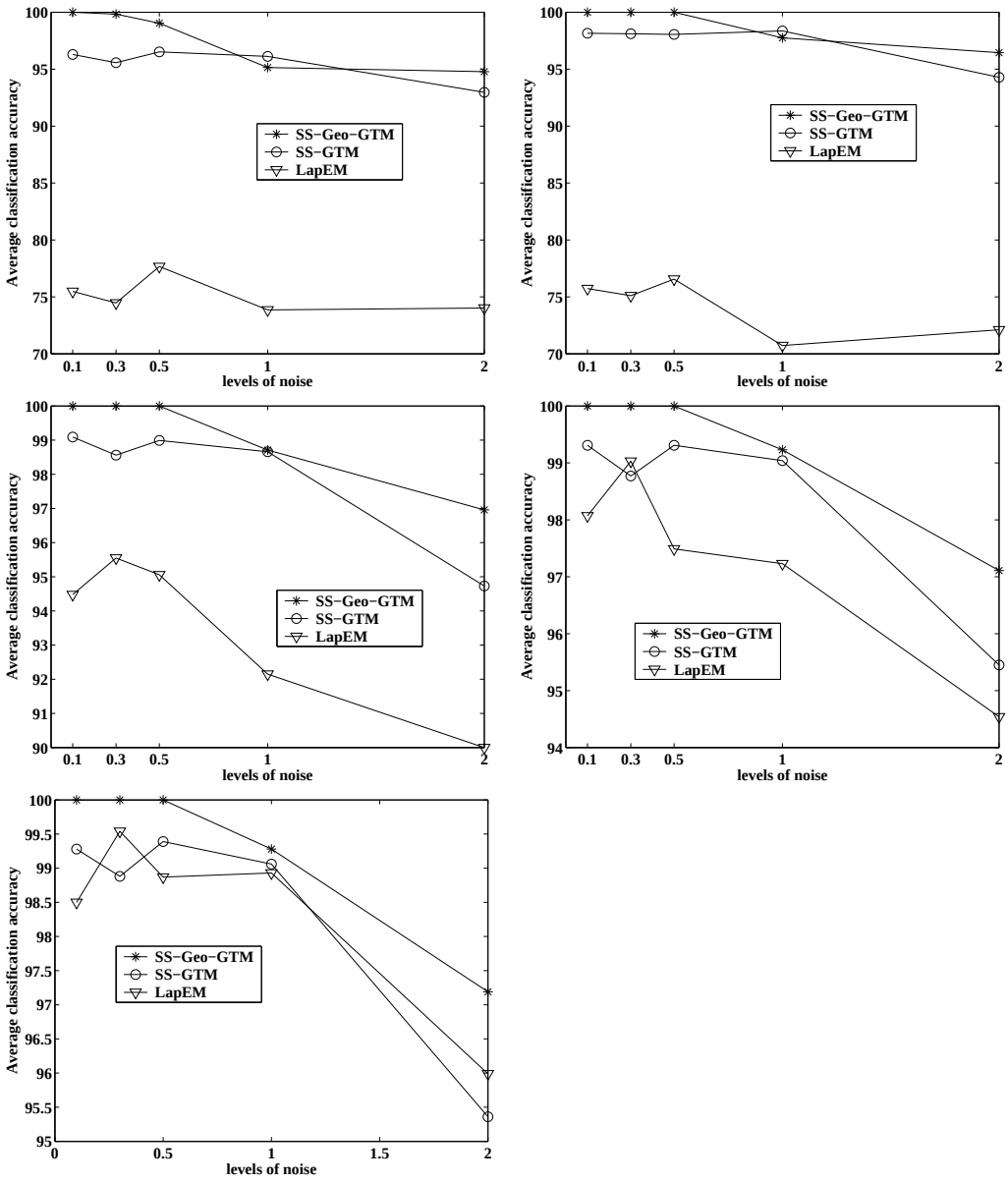

Figure 6.5: Average classification accuracy results taken from Table 6.7 using different and increasing percentage of labels per class for *Dalí* set. From left to right and from top to bottom, percent of labels $\% = 2, 4, 6, 8, 10$.

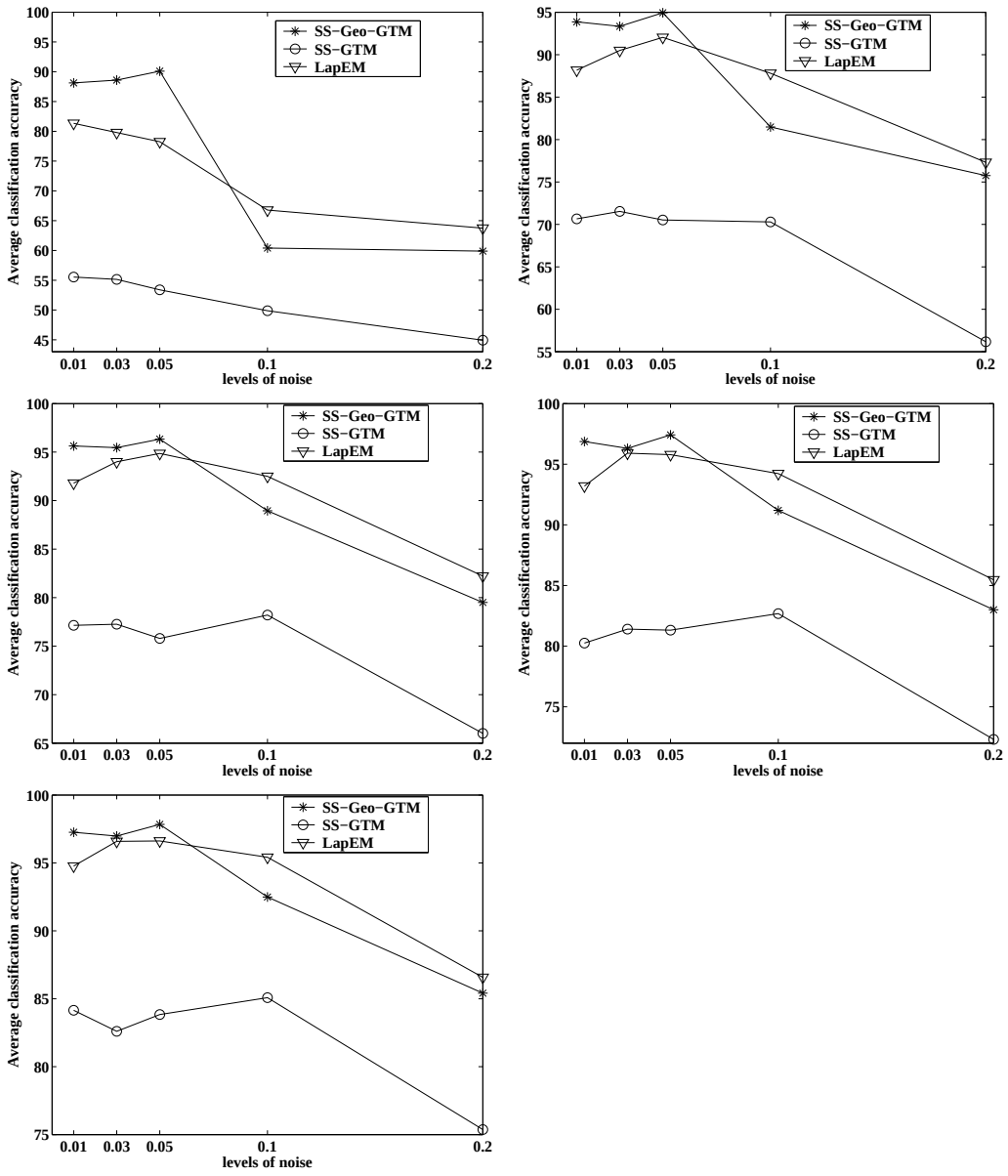

Figure 6.6: Average classification accuracy results taken from Table 6.7 using different and increasing percentage of labels per class for *Oil-Flow* set. From left to right and from top to bottom, percent of labels $\% = 2, 4, 6, 8, 10$.

cate that the resilience of the models is mostly due to the inclusion of the geodesic metric and not to the semi-supervised procedure itself. It is worth noting that the results for LapEM only become comparable as the percentage of available labels increases.

For a better appreciation of the results in Table 6.7, four figures (Fig. 6.3 to Fig. 6.6) were elaborated from it. In each of the graphics in Figs. 6.3 and 6.4 (for, in turn, the *Dalí* and *Oil-Flow* sets), the average classification accuracy for all models as a function of the percentage of available labels is shown for a specific level of noise. Likewise, in each of the graphics in Figs. 6.5 and 6.6, the average classification accuracy for all models as a function of the level of noise is shown for a specific percentage of available labels. From the first two figures, the advantage of SS-Geo-GTM in a convoluted but smooth dataset such as *Dalí* is clear at low levels of noise, and for small percentages of available labels; it is less so once we reach a 10% of label availability. For high levels of noise, the use of the geodesic metric becomes less relevant. With the less smooth and more heterogeneous *Oil-Flow* set[1], Laplacian Eigenmaps behave more robustly at higher noise level. The latter pair of figures provide a clear evidence of the differential and excellent behaviour of the label propagation procedure in SS-Geo-GTM in the most extreme conditions of label availability.

6.5 Experiments on a Human Brain Tumour Dataset

In this section, we leave artificial datasets and turn our attention to the analysis of a real problem in the field of biomedical applications. The problem of inferring survival stages in the development of an aggressive human brain tumour pathology is considered here from a semi-supervised point of view. This is a hard

[1] From Svensén (1998), we know that one of the classes in this data set, namely the stratified configuration flow, is discontinuous in nature and, therefore, its data are distributed over several separate manifolds. For this reason, a model that pays attention to local manifolds, such as Laplacian Eigenmaps, should in theory have at least a partial advantage in the task of modelling these data. Yet, in practice, SS-Geo-GTM is shown to perform overall better at low levels of noise.

problem because only a very limited amount of survival stage labels is available. It is also uncertain in its outcome because inference is based on the use of Magnetic Resonance Spectroscopy (MRS) data corresponding to these tumours (A description of the automated protocol used for the acquisition of these data can be found in Tate *et al.* 2003). To date, there is very limited evidence supporting the ability of MRS data to predict the survival of patients suffering from aggressive brain tumours. The analyzed pathology, namely glioblastomas, is known for its heterogeneity. As mentioned in the introduction to this chapter, to the best of our knowledge, this approach to survival stage analysis has never been attempted before using this type of data.

The data[1] used in this study consist of 86 single voxel PROBE (PROton Brain Exam system) MRS corresponding to glioblastomas: an aggressive type of brain tumour. They are a subset of the data described in section 5.4. For the analyses in this study, the available survival information of a patient was used as class labels. This corresponds to three stages describing the following outcomes, three months after diagnosis: "Deficits not impairing work or leisure" (8 cases), "Dependent but conscious" (11) and "Dead" (11). Notice that this means that less than 35% of the class labels are available.

The clinically relevant regions of the MRS were sampled to obtain 195 frequency intensity values. Given the scarcity of MRS data and their high dimensionality, the reported analyses do not resort to the full set of 195 spectral frequencies, but to a selection of 11 frequencies (as in Romero *et al.* 2009), known to be relevant for the characterization of aggressive tumours.

6.5.1 Experimental Design and Settings

Two problems are separately considered: The first is the discrimination between the "Deficits not impairing work or leisure" and the "Dead" prognostic stages;

[1]We gratefully acknowledge the members of the GABRMN research group at UAB, Barcelona, Spain, as well as former members of the INTERPRET European research project, for making these data available to us. We specifically thank Prof. Carles Arús and Dr. Margarida Julià-Sapé at GABRMN for supervising the medical quality of the research and for their valuable insights in the interpretation of MRS data.

the second is the discrimination between "Dependent but conscious" and "Dead". For none of these separate problems we have more of 26% of the class labels.

Geo-GTM, SS-Geo-GTM, and SS-GTM were all initialized following a procedure described in Bishop *et al.* (1998). The grid of latent points $\mathbf{u}_m$ was fixed to a square 7×7 layout. The corresponding grid of basis functions ϕ was equally fixed to a 5×5 square layout. We assume that the choice of the MRIP, described in section 6.2.1, as a value for σ is appropriate (as shown in section 6.3.2 and in Cruz-Barbosa & Vellido 2009a).

Given that there is no way for us to assess the performance of the models on the unavailable prognostic stage labels, we instead proceed in our experiments by subtracting a certain percentage of the available class labels from each dataset. Then, we aim to retrieve them, in a semi-supervised fashion, using SS-Geo-GTM, SS-GTM and the alternative method of Laplacian Eingenmaps (LapEM, Belkin & Niyogi 2003b). Only this way we can gauge the potential of the methods when faced with a real task of inferring the unavailable labels, that is, of inferring the prognostic stage.

In this setting, we evaluate SS-Geo-GTM, SS-GTM and LapEM models as follows: We subtract from 30 to 90 percent (percentages are rounded to the nearest whole number) of the available class labels from each class in each dataset, i.e., we only use from 70 to 10 percent of them. The remaining samples are considered as unlabeled data. The performance criterion is the retrieval accuracy of labels defined as the ratio of correctly retrieved labels to the total number of retrievable labels.

6.5.2 Results and Discussion

The two problems described in the previous section are first modeled using GTM and Geo-GTM. SS-GTM and SS-Geo-GTM are then built on top of these. As mentioned in the previous section, the randomly selected (from the total of available labels) labeled data per class is decreased from a 70% to a 10%. The remaining samples, both the corresponding to the subtracted labels and those without assigned survival type, are considered as unlabeled data. The semi-supervised

% of eliminated labels	**Retrieval accuracy** (ratio)		
	Problem 1		
	SS-Geo	SS-GTM	LapEM
30	5.03/7	3.38/7	3.51/7
40	5.74/9	4.41/9	5.09/9
50	6.42/10	4.87/10	5.87/10
60	7.04/12	6.08/12	7.21/12
70	8.10/14	7.04/14	8.43/14
80	9.19/16	8.36/16	9.56/16
90	9.34/17	8.54/17	9.11/17

Table 6.8: Average retrieval accuracy (as a ratio) over one hundred runs, for SS-GeoGTM, SS-GTM and LapEM. A randomly increasing percentage of pre-labeled items per class was chosen to be eliminated in each run.

performance of the models is measured as the average ratio of number of retrieved labels to the total number of labels to be retrieved (not including those without assigned survival type) over one hundred runs.

The corresponding results for the first problem ("Deficits not impairing work or leisure" vs. "Dead") are shown in Table 6.8. In the easiest case - the row corresponding to the 30% of eliminated labels (i.e. 70% of available labels out of the originally labeled data, or just over 15% of available labels out of all data)-, we observe that SS-Geo-GTM clearly outperforms the other models. In the most extreme case, when 90% of the available labels are eliminated (only 10% are kept labeled, which corresponds to about a 2% of cases overall) similarly poor results are obtained for all analysed models. Here, the complexity of the problem, in terms of the scarcity of spectra and the extreme paucity of class information seriously damage the performance of the models.

The results for the second problem ("Dependent but conscious" vs. "Dead") are shown in Table 6.9. They are overall similar to the results reported in Table 6.8. Here, SS-Geo-GTM outperforms SS-GTM and LapEM almost for all cases. The differences between the results obtained for SS-Geo-GTM and SS-GTM are never too high. This could mean that the analysed human brain tumour data do

% of eliminated labels	**Retrieval accuracy** (ratio)		
	Problem 2		
	SS-Geo	SS-GTM	LapEM
30	5.73/8	4.89/8	3.67/8
40	7.10/10	6.32/10	4.91/10
50	8.06/12	8.05/12	5.71/12
60	9.71/14	8.80/14	6.95/14
70	9.92/16	9.58/16	7.88/16
80	10.51/18	9.78/18	8.57/18
90	10.69/20	10.90/20	9.43/20

Table 6.9: Average retrieval accuracy (as a ratio) over one hundred runs, for SS-GeoGTM, SS-GTM and LapEM. A randomly increasing percentage of pre-labeled items per class was chosen to be eliminated in each run.

not present an excessively convoluted curvature. When there is no discrepancy between the graph and the Euclidean distance in the models (that is, when there is little convolution in the manifold), Geo-GTM tends to behave similarly to the standard GTM.

The analysed brain tumour MRS dataset has shown, overall, only a limited capability to inform the regeneration of missing labels through the proposed semi-supervised learning models. These results should therefore be used with extreme caution as an indicator of certain power to infer prognostic stages in human brain tumours (which is not to be underestimated, given that in real clinical settings there is a very limited posibility to offer aggresive brain tumour prognosis on the basis of MRS information). In any case, SS-Geo-GTM consistently outperformed the alternative methods.

6.6 Summary

A semi-supervised version of Geo-GTM, namely SS-Geo-GTM, has been theoretically defined and experimentally evaluated in this chapter. The main goal is inferring the unavailable class labels using the information provided by the few

available ones, as well as by the cluster structure defined by Geo-GTM. This model makes use of its defined prototypes as nodes in a proximity graph, where the edges are obtained using graph distances as approximation of the geodesic metric. From this setting, a modified class label propagation algorithm performs the semi-supervised task. Information derived from the training of Geo-GTM is used to derive a criterion (MRIP) for the selection of the σ parameter in the modified LP algorithm.

Through several experiments, the performance of SS-Geo-GTM has been assessed and it has been shown to be consistently better than that of the semi-supervised version of the standard GTM trained using the Euclidean metric, even in the presence of high levels of noise. Its performance has also been compared to that of LapEM in several synthetic datasets. It has been shown that SS-Geo-GTM significantly outperforms LapEM for all data sets and noise levels, with few exceptions. Further experiments concerning real data corresponding to the biomedical problem of survival stage inference using MRS information have been designed and carried out. Although the results are, overall, far from optimal, they are still useful and SS-Geo-GTM consistently outperformed the alternative methods.

As in the previous chapter, we must note that there is a limitation in the proposed models in the case of data multi-modality. As we introduce class information in the GTM and Geo-GTM modeling process, data multi-modality, if existing, is likely to generate discordance between the grouping or clustering structure of the data by itself and the distribution of classes, affecting the semi-supervised results, which are based on assumptions of class continuity. It must be noted, though, that the proposed semi-supervised Geo-GTM still performs quite well in a dataset likely to be affected by multi-modality, such as *Oil-Flow*, even when compared with Laplacian Eigenmaps, which is a model defined with locality in mind.

6. SEMI-SUPERVISED GEODESIC GENERATIVE TOPOGRAPHIC MAPPING

Part IV

Conclusion

Chapter 7

Conclusion

This chapter wraps up the thesis providing the reader with a summary of all its main developments. Its first section provides and overview of the main contributions of the previous chapters. This is followed by an assessment on the main novelties in this contributions and by an outline of some perspectives for future research that could build on this body of work.

7.1 Thesis Overview

The ultimate goal of this thesis was the development of novel generative manifold learning methods for the exploration of partially labeled data. Uncompletely labeled data sets are common in many of the databases generated in some of the currently most active areas of research, such as, for instance, biomedicine, bioinformatics, or web mining. That is one of the reasons why research on semi-supervised learning has steeply increased over the past few years. This thesis has indeed been driven by the same motivation. From the point of view of the application of the developed methods, we were specifically interested in the analysis of data corresponding to a diagnostic assistance problem in the field of oncology of human brain tumours. This interest is reflected in several chapters of this document.

Also, manifold learning methods research has attracted much attention of late because of their ability to model high-dimensional multivariate data under the assumption that these can be faithfully represented by a low-dimensional

manifold embedded in the observed data space. Again, this high dimensionality is increasingly common in datasets resulting from real-world problems. It is usually accompanied by a limitation in the interpretability of the experimental results, which can be alleviated by dimensionality reduction methods. In particular, in this thesis we used manifold learning methods for dimensionality reduction that stem from the generative approach.

At the beginning of this thesis, in part I, chapter 2, the state of the art on semi-supervised learning was presented. Also in this part, the foundations of generative manifold learning were described in chapter 3. The baseline method: Generative Topographic Mapping (GTM), which belongs to the manifold learning family, was described as well. GTM, with its focus on interpretability through visualization and clustering, has been shown to have several practical advantages over general finite mixture models.

In part II, chapter 4, a first exploration of a novel generative manifold learning method without assistance of class information was presented. Here, the Geodesic Generative Topographic Mapping (Geo-GTM) method was introduced. The Geo-GTM is an extension of GTM developed to favour the similarity of points along the learned manifold, while penalizing the similarity of points that are not contiguous in the manifold, even if close in terms of the Euclidean distance. This was achieved by modifying the standard calculation of the responsibilities in Eq. 3.4 in proportion to the discrepancy between the geodesic (approximated by the graph) and the Euclidean distances. Geo-GTM was shown in this chapter to be able to faithfully recover and visually represent the underlying structure of datasets of smooth but convoluted geometries. The reported experiments also showed that Geo-GTM was capable of recovering the true underlying data structure far better than the standard GTM, even in the presence of a considerable amount of noise.

Two models of the generative manifold learning family using class information were explored in part III of the thesis. A first exploration using labeled data in a two-stage clustering method was presented in chapter 5. A variation on class-GTM model was developed to assist a two-stage clustering procedure in this chapter. Class-GTM is an extension of GTM, where the main goal of this extension is to improve class separability in the clustering results of GTM. In

the two-stage clustering procedure proposed in this chapter, the first stage involves class-GTM and the second stage involves K-means. Also, two novel fixed initialization strategies that take advantage of the prior knowledge obtained by class-GTM in the first stage were introduced. They are based on two features of the model, namely: the Magnification Factors (MF) and the Cumulative Responsibility (CR). Several experiments in this chapter showed that the two-stage random and fixed initializations yield almost identical results in terms of clusterwise class separation, with the former being computationally more costly. It was also shown that the two-stage clustering procedures based on class-GTM perform much better than direct K-means and GTM clustering of the data in terms of this clusterwise class separation. The existence of atypical data or outliers in a human brain tumours MRS dataset, and its influence on the clustering process, were also explored in this chapter. For this last analysis, the class-t-GTM model was developed.

A more powerful model, in terms of items' labeling capability, was developed in chapter 6. Here, the Semi-Supervised Geodesic Generative Topographic Mapping (SS-Geo-GTM) model was introduced. The basic idea underlying the proposed semi-supervised approach is that neighbouring points are most likely to share their label and that these labels are best propagated through neighbouring nodes according to proximity. Unlike models in chapter 5, SS-Geo-GTM was designed for classification tasks. For this purpose, an existing label propagation (LP) algorithm was modified to account for the information provided by the trained Geo-GTM (previously developed in chapter 4). We profited from experiments carried out in chapter 5 (in which clear indications were found suggesting that the prototypes with highest CR could be considered as the most representative in the dataset), as well as from the information provided by Geo-GTM training, in order to conclude that the proposed main reference inter-prototype (MRIP) distance was an appropriate value for the parameter σ of the corresponding modified LP algorithm. In a similar way, a semi-supervised version of the standard GTM (SS-GTM) was developed. Through several experiments, the performance of SS-Geo-GTM, in terms of classification accuracy, was assessed and shown to be consistently better than that of the semi-supervised version of the standard GTM trained using the Euclidean metric, even in the presence of high levels of

noise. Its performance was also compared to that of LapEM in several synthetic datasets and shown that it significantly outperformed LapEM for all datasets and noise levels, with few exceptions. Further experiments concerning real data corresponding to the biomedical problem of survival stage inference, using MRS information, were designed and carried out. Although the results are preliminary and not completely satisfactory, they are still useful in medical terms and SS-Geo-GTM consistently outperformed the alternative methods.

7.2 Impact of the Main Contributions

The major contributions of this thesis are highlighted in this section. The first one is the definition of Geo-GTM as a principled extension of GTM to uncover underlying structures in convoluted datasets. The second one is the definition of a two-stage clustering procedure as an extension of GTM. The third one is the definition of SS-Geo-GTM as a principled extension of Geo-GTM to semi-supervised problems. The last one is the novel application of semi-supervised models of the manifold learning family to the assistance of exploratory unsupervised clustering of real NMR spectroscopy data with uncertain prognostic labeling.

7.2.1 Geodesic Generative Topographic Mapping

In chapter 4, we define Geo-GTM as a principled extension of GTM to uncover underlying structures in convoluted datasets, by explicitly penalizing the differences between the Euclidean and the alternative geodesic distance from data to prototypes in the original constrained mixture model. This penalization helps to alleviate, in part, the trustworthiness and continuity problems defined in section 4.4.1, which are particularly common in convoluted datasets. Some examples of results on this kind of datasets are presented in section 4.4.1. Also Geo-GTM, by definition, can represent low dimensional manifolds of smooth curvature as well as GTM does.

For its definition and characteristics, Geo-GTM can be used for clustering and visualization analysis in many real application areas datasets. Since the intrinsic high dimensionality of this kind of data does not allow to directly visualize them,

we can apply GTM and Geo-GTM models to determine whether it presents convoluted curvature. When the results of both models are similar, it means that the dataset can be represented by a low dimensional manifold of smooth curvature, otherwise it presents convoluted geometry. Some examples corroborating this procedure are (implicitly) presented in chapter 6.

7.2.2 Two-stage Clustering with class-GTM

In chapter 5, we define a two-stage clustering procedure as a principled extension of GTM by explicitly using class-GTM in the first stage and K-means in the second one. That is, class labels are used to enrich and refine the cluster structure discovered in a two-stage clustering process. Class-GTM model integrates class information as part of the GTM training to enrich the cluster structure definition provided by the model. For visualization purposes, Class-GTM do not place any strong restriction on the number of mixture components (or clusters, which exploits the substructure in the input data). This mixture of components does not necessarily match the more global cluster and class distributions of the data. Thus, the centres of these components, also known as prototypes, are further clustered in the second stage by using K-means. In this way, we explore the use of class information in unsupervised clustering, which is a far less frequently investigated problem in comparison with supervised classification problems.

Two novel initialization procedures for the second stage (K-means), derived from class-GTM training, are also defined in chapter 5. These fixed initialization strategies are based on two features of the class-GTM model, namely: the Magnification Factors (which measure the level of stretching that the mapping undergoes from the latent to the data spaces) and the Cumulative Responsibility (which is the sum of responsibilities over all data points for each cluster). Then, making use of the prior knowledge generated by class-GTM and without compromising the final clusterwise class separation capabilities of the model, these fixed initialization procedures allow significant computational savings compared with a random initialization procedure, as showed in section 5.3.2. The assumption that the prototypes with highest Cumulative Responsibility are the most representative in the dataset is used by taking the prototypes with highest CR as seeds

for the initialization of K-means. Obviously, this technique might be used with other clustering methods requiring seeds for its initialization procedure. Also, this result was a key factor to select the suitable value for parameter σ in chapter 6.

7.2.3 Semi-Supervised Geodesic Generative Topographic Mapping

In semi-supervised problems, only a reduced number of class labels is readily available and even those could be difficult and/or expensive to obtain. This scenario es very common in many of the databases generated in some of the currently most active areas of research, such as, for instance, biomedicine, bioinformatics, or web mining.

A principled extension of Geo-GTM to semi-supervised problems is defined as SS-Geo-GTM in chapter 6 by explicitly introducing a modified label propagation algorithm on top of Geo-GTM. The classification task is the purpose of the SS-Geo-GTM model, but using a clustering method (Geo-GTM) as a basis. That is, the resulting SS-Geo-GTM uses the information derived from Geo-GTM training to accomplish the semi-supervised task. In this sense, the MRIP criterion is proposed to select the suitable value for σ in the modified LP algorithm. Experimental results showed that MRIP is near the optimal value for σ.

The results presented in chapter 6 show that SS-Geo-GTM outperformed SS-GTM and the alternative LapEM method when applied to datasets with differently convoluted geometries. Further experiments using different percentages of available class labels and also with the presence of different levels of uninformative noise, show that SS-Geo-GTM overall outperforms both SS-GTM and LapEM. Thus, through this contribution (SS-Geo-GTM model), researchers on semi-supervised learning could explore the areas mentioned at the beginning of this section, as illustrated by sections 6.4.2 and 6.5.2 of the thesis.

7.2.4 Analysis of a Human Brain Tumour Dataset using Class Information

A first analysis of the human brain tumour dataset, described in section 5.4, is presented in section 5.4.2. The data in this study are spectra obtained through Nuclear Magnetic Resonance spectroscopy. These spectra, by themselves, are difficult to obtain, standardize and preprocess for analysis. This study is based on a two-stage clustering procedure. In section 5.4.2, the results show that the inclusion of class information improves the clusterwise tumour type separation both using the fourteen tumour types and using three important groups of them. Also, the existence of atypical data or outliers in the datasets under study, and their influence on the clustering process was explored. This analysis concluded that not too many data could be clearly characterized as outliers, most of them belonging to the best represented tumour types in the dataset. These kinds of presented analyses could be used as a guide to apply with other MRS datasets.

In chapter 6, a second study using human brain tumour spectra obtained through NMR spectroscopy is presented. This time, we evaluated the capability of SS-Geo-GTM model in the difficult problem of inferring survival stages in an aggressive human brain tumour pathology from a very limited amount of available survival stage labels and Magnetic Resonance Spectroscopy data corresponding to these tumours. Given that there was no way for us to assess the performance of the models on the unavailable prognostic stage labels, we instead proceeded in our experiments by subtracting a certain percentage of the available class labels from each dataset. Then, we aimed to retrieve them, in a semi-supervised fashion, using SS-Geo-GTM, SS-GTM and the alternative method of LapEM. The results in the analysed brain tumour MRS dataset show, overall, only a limited capability to inform the regeneration of missing labels through the proposed semi-supervised learning models. Although the results are, overall, far from optimal, they are still useful (which is not to be underestimated, given that in real clinical settings there is a very limited posibility to offer aggresive brain tumour prognosis on the basis of MRS information).

7.3 Future Work

Some novel models, contributing to the fields of data clustering and visualization and semi-supervised learning have been presented in this thesis. From here, there exist several research avenues open to exploration. Next, we suggest some possible extensions on the body of this work.

All of the manifold learning models this thesis has dealt with are likely to struggle in the modelling of high-dimensional sparse datasets. This includes GTM itself and the proposed Geo-GTM, as well as their semi-supervised counterparts, SS-GTM and SS-Geo-GTM. This is because the own geometric properties of sparsely populated high-dimensional data spaces are likely to influence the data modelling process very strongly. This problem could be alleviated by using an approach similar to that presented in Kabán (2005) to deal with sparse data sequences. This approach is feasible with Geo-GTM, in the sense that the divergences between the Euclidean distances from the data points to the model prototypes or means (in this case) and the corresponding approximated geodesic distances along the manifold could still be penalized in the E-step of the E-M algorithm proposed in Kabán (2005).

Moving now to SSL, semi-supervised clustering is a challenging and not too investigated problem in this field. In this context, a very interesting task is semi-supervised clustering using only constraints information instead of cluster labels. Semi-supervised clustering with pairwise constraints emerges because it can be a more natural form of supervision than labels in certain clustering tasks. In this approach, pairwise supervision is typically provided as a *must-link* constraint on data points (indicating that both points in a pair should be placed in the same cluster) or a *cannot-link* constraint (indicating that both points in a pair should belong to different clusters). In this sense, a semi-supervised extension of Geo-GTM (different but following some ideas from that defined in chapter 6) using pairwise constraints might be defined in the following way. The pairwise constraints information of data points could be used to fix the pairwise constraints of Geo-GTM prototypes to which they are assigned. Then, The Geo-GTM prototypes and their corresponding pairwise constraints could be adapted and used as inputs for the HMRF-KMeans algorithm proposed in Basu (2005).

As acknowledged in previous chapters, the models developed in this thesis (as well as many other manifold learning models) implicitly assume the intrinsic continuity of the data. This implies a limitation when dealing with multi-modal data. An approach to deal with such limitation that could be used in future research would entail the definition of hierarchical extensions of Geo-GTM and of its semi-supervised counterpart. They would allow dealing with unconnected regions sharing a common data generator. This would be inspired in hierarchical extensions of GTM such as those defined in Tiňo & Nabney (2002) and Nabney *et al.* (2005).

Bibliography

Archambeau, C. & Verleysen, M. (2005). Manifold constrained finite gaussian mixtures. In J. Cabestany, A. Prieto & D.F. Sandoval, eds., *Proceedings of IWANN*, vol. LNCS 3512, 820–828, Springer-Verlag. 32, 34, 35

Asuncion, A. & Newman, D. (2007). UCI machine learning repository [http://www.ics.uci.edu/∼mlearn/MLRepository.html]. University of California, Irvine, School of Information and Computer Sciences. 80

Aupetit, M. (2003). Robust topology representing networks. In *Proceedings of the 11th European Symposium on Artificial Neural Networks (ESANN 2003)*, 45–50, d-side. 33

Basu, S. (2005). *Semi-supervised Clustering: Probabilistic Models, Algorithms and Experiments*. Ph.D. thesis, The University of Texas at Austin, U.S.A. 11, 112

Belkin, M. & Niyogi, P. (2003a). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, **15**, 1373–1396. 20

Belkin, M. & Niyogi, P. (2003b). Using manifold structure for partially labelled classification. In *Advances in Neural Information Processing Systems (NIPS)*, vol. 15, MIT Press. 74, 87, 88, 98, 129

Belkin, M. & Niyogi, P. (2004). Semi-supervised learning on Riemannian manifolds. *Machine Learning*, **56**, 209–239. 13, 21, 74

Bernstein, M., de Silva, V., Langford, J. & Tenenbaum, J. (2000). Graph approximations to geodesics on embedded manifolds. Tech. rep., Stanford University, CA, U.S.A. 33, 129

BIBLIOGRAPHY

Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press. 14

Bishop, C.M. (1999). Latent variable models. In M.I. Jordan, ed., *Learning in Graphical Models*, 371–403, MIT Press. 21

Bishop, C.M. & James, G.D. (1993). Analysis of multiphase flows using dual-energy gamma densitometry and neural networks. *Nuclear Instruments and Methods in Physics Research A*, **327**, 580–593. 56

Bishop, C.M., Svensén, M. & Williams, C.K.I. (1997). Magnification Factors for the GTM algorithm. In *Proceedings of the IEE fifth International Conference on Artificial Neural Networks*, 64–69. 52, 53

Bishop, C.M., Svensén, M. & Williams, C.K.I. (1998). The Generative Topographic Mapping. *Neural Computation*, **10**, 215–234. 20, 21, 31, 34, 35, 54, 74, 80, 88, 98, 126

Blum, A. & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory (COLT 98)*, 92–100. 4, 10

Bouchard, G. & Triggs, B. (2004). The trade-off between generative and discriminative classifiers. In *IASC 16th International Symposium on Computational Statistics*, 721–728. 17

Brand, M. (2003). Charting a manifold. In *Advances in Neural Information Processing Systems*, 857–864. 20

Burges, C.J.C. & Platt, J.C. (2006). Semi-supervised learning with conditional harmonic mixing. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 13

Chapelle, O., Schölkopf, B. & Zien, A., eds. (2006). *Semi-Supervised Learning*. The MIT Press. 4, 11, 12, 13, 21, 73

Cozman, F. & Cohen, I. (2006). Risks of semi-supervised learning. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 13

Cruz-Barbosa, R. & Vellido, A. (2006). On the improvement of brain tumour data clustering using class information. In *Proceedings of the 3rd European Starting AI Researcher Symposium (STAIRS'06), Riva del Garda, Italy*. 6, 50, 51, 62

Cruz-Barbosa, R. & Vellido, A. (2007a). Evaluation of a two-stage clustering procedure using class information in Generative Topographic Mapping. In I. Rojas-Ruiz & H. Pomares-Cintas, eds., *Actas del II Simposio de Inteligencia Computacional (IEEE SICO 2007)*, 17–24, Thomson. 6

Cruz-Barbosa, R. & Vellido, A. (2007b). Limits to the use of class information in a GTM-based two-stage clustering procedure. In F. Ferrer-Troyano, A. Troncoso & J. Riquelme, eds., *Actas del IV Taller Nacional de Minera de Datos y Aprendizaje (TAMIDA 2007)*, 303–312, Thomson. 6

Cruz-Barbosa, R. & Vellido, A. (2007c). On the influence of class information in the two-stage clustering of a human brain tumour dataset. In A. Gelbukh & A. Kuri-Morales, eds., *Proceedings of the 6th Mexican Conference on Artificial Intelligence (MICAI 2007)*, vol. 4827 of *LNAI*, 472–482, Springer. 6, 77

Cruz-Barbosa, R. & Vellido, A. (2007d). On the initialization of two-stage clustering with class-GTM. In D. Borrajo, L. Castillo & J. Corchado, eds., *Proceedings of the 12th Conference of the Spanish Association for Artificial Intelligence, CAEPIA+TTIA 2007*, vol. 4788 of *LNAI*, 50–59, Springer. 6, 77

Cruz-Barbosa, R. & Vellido, A. (2008a). Geodesic Generative Topographic Mapping. In H. Geffner, R. Prada, I. Alexandre & N. David, eds., *Proceedings of the 11th Ibero-American Conference on Artificial Intelligence (IBERAMIA 2008)*, vol. 5290 of *LNAI*, 113–122, Springer. 6, 74, 81, 84, 87, 89

Cruz-Barbosa, R. & Vellido, A. (2008b). On the improvement of the mapping trustworthiness and continuity of a manifold learning model. In C. Fyfe, D. Kim, S.Y. Lee & H. Yin, eds., *Proceedings of the 9th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2008)*, vol. 5326 of *LNCS*, 266–273, Springer. 6, 74

Cruz-Barbosa, R. & Vellido, A. (2008c). Two-stage clustering of a human brain tumour dataset using manifold learning models. In *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing, BIOSIGNALS 2008*, 191–196, INSTICC Press. 6

Cruz-Barbosa, R. & Vellido, A. (2008d). Unfolding the manifold in Generative Topographic Mapping. In E. Corchado, A. Abraham & W. Pedrycz, eds., *Proceedings of the 3rd International Workshop on Hybrid Artificial Intelligence Systems (HAIS 2008)*, vol. 5271 of *LNAI*, 392–399, Springer. 6, 74

Cruz-Barbosa, R. & Vellido, A. (2009a). Comparative evaluation of semi-supervised Geodesic GTM. In E. Corchado et al., ed., *Proceedings of the 4th International Conference on Hybrid Artificial Intelligence Systems (HAIS 2009)*, vol. 5572 of *LNAI*, 344–351, Springer. 7, 98

Cruz-Barbosa, R. & Vellido, A. (2009b). Semi-supervised analysis of a human brain tumour type using survival infomation. *Submitted to the 10th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2009)*. 7

Cruz-Barbosa, R. & Vellido, A. (2009c). Semi-supervised geodesic Generative Topographic Mapping. *Submitted to Pattern Recognition Letters journal*. 7

Davies, D.L. & Bouldin, D.W. (1979). A cluster separation measure. *IEEE Trans. on Pattern Analysis and Machine Intelligence,*, **1**, 224–227. 55

de Silva, V. & Tenenbaum, J. (2003). Unsupervised learning of curved manifolds. In D. Denison, M. Hansen, C. Holmes, B. Mallick & B. Yu, eds., *Nonlinear Estimation and Classification, Lecture Notes in Statistics*, vol. 171, 453–466, Springer Verlag, New York. 21

Dempster, A.P., Laird, N.M. & Rubin, D.B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society B*, **39**, 1–38. 15, 23

Dijkstra, E.W. (1959). A note on two problems in connection with graphs. *Numerische Mathematik*, **1**, 269–271. 33, 130

Duda, R.O., Hart, P.E. & Stork, D.G. (2000). *Pattern Classification*. Wiley-Interscience, 2nd edition. 52

Figueiredo, M.A.T. & Jain, A.K. (2002). Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 381–396. 49

Ghahramani, Z. & Jordan, M.I. (1994). Supervised learning from incomplete data via the EM approach. In *Advances in Neural Information Processing Systems*, 6, 120–127. 4, 10

Golub, G.H. & Reinsch, C. (1970). Singular value decomposition and least squares solutions. *Numerische Mathematik*, **14**, 403–420. 20

Grandvalet, Y. & Bengio, Y. (2006). Entropy regularization. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 13

Hastie, T. & Stuetzle, W. (1988). Principal curves. *Journal of the American Statistical Association*, **84**, 502–516. 20

Hastie, T. & Tibshirani, R. (1996). Discriminant analysis by Gaussian mixtures. *Journal of the Royal Statistical Society (B)*, **58**, 155–176. 50

Herrmann, L. & Ultsch, A. (2007). Label propagation for semi-supervised learning in self-organizing maps. In *Proceedings of the 6th WSOM 2007*. 74, 76

Hinton, G. & Roweis, S. (2003). Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems*, 857–864. 20

Jain, A.K. & Dubes, R.C. (1998). *Algorithms for Clustering Data*. Prentice Hall, New Jersey. 3, 11, 73

Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML-99)*, 200–209. 4, 10

Jolliffe, I.T. (2002, 2nd edition). *Principal Component Analysis*. Springer Series in Statistics, Springer Verlag. 20

Kabán, A. (2005). A scalable generative topographic mapping for sparse data sequences. In *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05)*, IEEE Computer Society. 112

Kaski, S., Sinkkonen, J. & Klami, A. (2005). Discriminative clustering. *Neurocomputing*, **69**, 18–41. 17

Kégl, B., Krzyzak, A., Linder, T. & Zeger, K. (2000). Learning and design of principal curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, 281–297. 20

Kohonen, T. (1995). *Self-Organizing Maps*. Springer-Verlag, Berlin. 22

Kruskal, J.B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. In *Proceedings of the American Mathematical Society*, vol. 7, 48–50. 130

Lasserre, J., Bishop, C.M. & Minka, T. (2006). Principled hybrids of generative and discriminative models. In *Proceedings 2006 IEEE Conference on Computer Vision and Pattern Recognition*. 17

Lawrence, N.D. & Jordan, M.I. (2006). Gaussian processes and the null-category noise model. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 13

Lee, J. & Verleysen, M. (2007). *Nonlinear Dimensionality Reduction*. Springer. 33, 129

Lee, J.A., Lendasse, A. & Verleysen, M. (2002). Curvilinear distance analysis versus isomap. In *Proceedings of European Symposium on Artificial Neural Networks (ESANN)*, 185–192. 32

Lisboa, P., Vellido, A. & Wong, H. (2000). Outstanding issues for clinical decision support with neural networks. In H. Malmgren, M. Borga & L. Niklasson, eds., *Artificial Neural Networks in Medicine and Biology*, 63–71, Springer, London. 59

McLachlan, G.J. & Basford, K.E. (1988). *Mixture Models: Inference and Applications to Clustering*. New York: Marcel Dekker. 15

Nabney, I.T., Sun, Y., Tiňo, P. & Kabán, A. (2005). Semisupervised learning of hierarchical latent trait models for data visualization. *IEEE Transactions on Knowledge and Data Engineering*, **17**, 384–400. 113

Nigam, K., McCallum, A., Thrun, S. & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 103–134. 4, 10

Nigam, K., McCallum, A. & Mitchell, T. (2006). Semi-supervised text classification using EM. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 12

Olier, I. & Vellido, A. (2008). Advances in clustering and visualization of time series using GTM Through Time. *Neural Networks*, **21**, 904–913. 32

Peltonen, J., Klami, A. & Kaski, S. (2004). Improved learning of Riemannian metrics for exploratory analysis. *Neural Networks*, **17**, 1087–1100. 17

Prim, R.C. (1957). Shortest connection networks and some generalizations. *Bell System Technical Journal*, 1389–1401. 130

Romero, E., Vellido, A., Julià-Sapé, M. & Arús, C. (2009). Discriminating glioblastomas from metastases in a SV ^{1}H-MRS brain tumour database. In *European Soc. for MR in Biology and Medicine (ESMRMB) Congress 2009. Submitted*. 97

Roweis, S.T. & Lawrence, K.S. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 2323–2326. 20

Saul, L.K., Weinberger, K.Q., Sha, F., Ham, J. & Lee, D.D. (2006). Spectral methods for dimensionality reduction. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 14

Seeger, M. (2000). Learning with labeled and unlabeled data. Tech. rep., Institute for ANC, Edinburgh, UK. 4, 10, 11, 16

Shin, H., Hill, N.J. & Rätsch, G. (2006). Graph based semi-supervised learning with sharper edges. In J. Fürnkranz, T. Scheffer & M. Spiliopoulou, eds., *Proceedings of the 17th European Conference on Machine Learning (ECML 2006)*, vol. 4212 of *LNAI*, 402–413, Springer-Verlag. 132

Sindhwani, V., Belkin, M. & Niyogi, P. (2006). The geometric basis of semi-supervised learning. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 13

Sun, Y., Tiňo, P. & Nabney, I.T. (2002). Visualization of incomplete data using class information constraints. In J. Winkler & M. Niranjan, eds., *Uncertainty in Geometric Computations*, 165–174, Kluwer Academic Publishers, The Netherlands. 50, 51

Svensén, M. (1998). *GTM: The Generative Topographic Mapping*. Ph.D. thesis, Aston University, U.K. 4, 21, 96, 126

Tate, A.R., Griffiths, J.R., Martínez-Pérez, I., Moreno, A., Barba, I., Cabanas, M.E., Watson, D., Alonso, J., Bartumeus, F., Isamat, F., Ferrer, I., Vila, F., Ferrer, E., Capdevila, A. & Arús, C. (1998). Towards a method for automated classification of ^{1}H MRS spectra from brain tumours. *NMR In Biomedicine*, **11**, 177–191. 3

Tate, A.R., Majós, C., Moreno, A., Howe, F.A., Griffiths, J.R. & Arús, C. (2003). Automated classification of short echo time in In Vivo ^{1}H brain tumor spectra: a multicenter study. *Magnetic Resonance in Medicine*, **49**, 29–36. 59, 97

Tate, A.R., Underwood, J., Acosta, D.M., Julià-Sapé, M., Majós, C., Moreno-Torres, A., Howe, F.A., van der Graaf, M., Lefournier, V., Murphy, M.M., Loosemore, A., Ladroue, C., Wesseling, P., Bosson, J.L., Cabañas, M.E., Simonetti, A.W., Gajewicz, W., Calvar, J., Capdevila, A., Wilkins, P.R., Bell, B.A., Rémy, C., Heerschap, A., Watson, D., Griffiths, J.R. & Arús, C. (2006). Development of a decision support system for diagnosis and grading of brain tumours using in vivo magnetic resonance single voxel spectra. *NMR in Biomedicine*, **19**, 411–434. 59

Tenenbaum, J.B., de Silva, V. & Langford, J.C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, **290**, 2319–2323. 19, 20, 21, 32

Titterington, D.M., Smith, A.F.M. & Makov, U.E. (1985). *Statistical Analysis of Finite Mixture Distributions*. New York: John Wiley. 15

Tiňo, P. & Nabney, I.T. (2002). Hierarchical GTM: Constructing localized nonlinear projection manifolds in a principled way. *IEEE Trnsactions on Pattern Analysis and Machine Inteligence*, **24**, 639–656. 113

Ultsch, A. (2003). Maps for the visualization of high-dimensional data spaces. In *Proceedings of WSOM 2003*, 225–230. 74

Vellido, A. (2006). Missing data imputation through GTM as a mixture of t-distributions. *Neural Networks*, **19**, 1624–1635. 24, 32, 60, 61

Vellido, A. & Lisboa, P.J.G. (2006). Handling outliers in brain tumour MRS data analysis through robust topographic mapping. *Computers in Biology and Medicine*, **36**, 1049–1063. 24, 62

Vellido, A., Lisboa, P.J.G. & Meehan, K. (2000). The Generative Topographic Mapping as a principled model for data visualization and market segmentation: an electronic commerce case study. *International Journal of Computers, Systems, and Signals*, **1**, 119–138. 52

Vellido, A., Lisboa, P. & Vicente, D. (2006). Robust analysis of MRS brain tumour data using t-GTM. *Neurocomputing*, **69**, 754–768. 32

Venna, J. (2007). *Dimensionality reduction for visual exploration of similarity structures*. Ph.D. thesis, Helsinki University of Technology. 31

Venna, J. & Kaski, S. (2001). Neighborhood preservation in nonlinear projection methods: An experimental study. In G. Dorffner, H. Bischof & K. Hornik, eds., *Proceedings of ICANN 2001*, 485–491, Springer, Berlin. 40

Vesanto, J. & Alhoniemi, E. (2000). Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*. 50, 52, 54, 55

Young, F. (1981). *Introduction to Multidimensional Scaling: Theory, Methods, and Applications*. Academic Press, New York. 20

Zhao, H. (2006). Combining labeled and unlabeled data with graph embedding. *Neurocomputing Letters*, 2385–2389. 132

Zhu, X. & Ghahramani, Z. (2002). Learning from labeled and unlabeled data with label propagation. Tech. rep., CMU-CALD-02-107, Carnegie Mellon University. 74, 76, 77, 127

Zhu, X., Kandola, J., Lafferty, J. & Ghahramani, Z. (2006). Graph kernels by spectral transforms. In O. Chapelle, B. Schölkopf & A. Zien, eds., *Semi-Supervised Learning*, The MIT Press. 14

Appendix A

Pseudocode for Some Algorithms Used in this Thesis

In this first appendix, two algorithms at the basis of some of the models used in the thesis are described. The first is the EM algorithm for GTM, which was modified to obtain the Geo-GTM model described in chapter 4. A second algorithm, label propagation, was modified and used with information derived from Geo-GTM to develop the SS-Geo-GTM model. This model is described in chapter 6.

A.1 EM algorithm for GTM

The steps for constructing a GTM model are:

- Generate the grid of latent points $\{\mathbf{u}_k\}, k = 1, \ldots, K$.
- Generate the grid of basis function centres $\{\mu_m\}, m = 1, \ldots, M$.
- Select the basis function width σ.
- Compute the matrix of basis function activations, $\mathbf{\Phi}$, from Eq. 3.2.
- Initialize $\mathbf{W}$, either randomly or with a fixed scheme (e.g., PCA-based).
- Initialize β, either randomly or with a fixed scheme (e.g., PCA-based).
- Compute $\mathbf{\Delta}$, $\Delta_{kn} = \|\mathbf{x}_n - \mathbf{\Phi}_k \mathbf{W}\|^2$.

- **repeat**
  - Compute $\mathbf{Z}$ from Eq. 3.4 using $\mathbf{\Delta}$ and β.
  - Compute $\mathbf{G}$ (a $K \times K$ diagonal matrix), where $g_{kk} = \sum_n^N z_{kn}$.

  } E − step

  - $\mathbf{W} = (\mathbf{\Phi^T G \Phi} + \lambda \mathbf{I})^{-1} \mathbf{\Phi^T Z X}$, where λ may be zero.
  - Compute $\mathbf{\Delta}$, $\Delta_{kn} = \|\mathbf{x}_n - \mathbf{\Phi}_k \mathbf{W}\|^2$.
  - update β according to $\beta^{-1} = \frac{1}{ND} \sum_n^N \sum_k^K z_{kn} \|\mathbf{y}(\mathbf{u}_k, \widetilde{\mathbf{W}}) - \mathbf{x}_n\|^2$.

  } M − step

- **until** convergence

Here, $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ is a set of data points, where each point $\mathbf{x}_n$ has dimension D.

For details on PCA-based initialization of $\mathbf{W}$ the reader is referred to Bishop *et al.* (1998) or Svensén (1998).

A.2 Label Propagation

The basic label propagation algorithm assumes that a set of data points $X = \{x_1, \ldots, x_{l+u}\}$ (where $x_i \in \mathbb{R}^D$) can be decomposed in available labeled data $(x_1, y_1), \ldots, (x_l, y_l)$ and unlabeled data $(x_{l+1}, y_{l+1}), \ldots, (x_{l+u}, y_{l+u})$. Here, $Y_L = \{y_1, \ldots, y_l\}$ and $Y_U = \{y_{l+1}, \ldots, y_{l+u}\}$ are, in turn, the observed and unobserved class labels, where, usually $l \ll u$. It is also assumed that the number of classes C is known and that all classes are present in the labeled data.

Some preprocessing is needed before the algorithm is run. Thus, we first introduce the preprocessing stage, and then the LP algorithm is described.

- Pre-processing stage
  - Create a fully connected graph where the nodes are all data points, both labeled and unlabeled.
  - Compute the weights of the edges between nodes i, j as: $w_{ij} = \exp\left(-\frac{\sum_{d=1}^D (x_i^d - x_j^d)^2}{\sigma^2}\right)$, where σ controls the weights of the edges.
  - Compute a $(l+u) \times (l+u)$ transition matrix T as $T_{ij} = \frac{w_{ij}}{\sum_{k=1}^{l+u} w_{kj}}$, where T_{ij} is the probability of propagation from node j to node i.

– Define a $(l+u) \times C$ label matrix Y, whose ith row represents the label probability distribution of node x_i. The initialization of unlabeled data points is not relevant.

Label propagation algorithm

1. Propagate $Y \leftarrow TY$
2. Row-normalize Y as $Y_{ij} = Y_{ij} / \sum_k Y_{ik}$.
3. Clamp the labeled data. Repeat from step 1 until Y converges.

For more details on this algorithm the reader is referred to Zhu & Ghahramani (2002).

A. PSEUDOCODE FOR SOME ALGORITHMS USED IN THIS THESIS

Appendix B

Graph Construction

This appendix provides further detail on the issue of graph construction on the basis of a given dataset. It is important to note that the way the graph is constructed may influence both the accuracy of the underlying structures revealed by the model and, therefore, the subsequent semi-supervised process.

As we have seen in section 4.2, the geodesic metric can be approximated by graph distances (Bernstein *et al.*, 2000). To be considered as a distance, the graph metric must comply with the conditions of a distance function as described in section 3.3. Given two data points lying on a manifold, the graph distance finds the shortest path between them, where such path is built by connecting the closest successive data points. This can be done using the K-rule, which allows connecting the K-nearest neighbors, or the ϵ-rule, which allows connecting data points x and y whenever $\|x - y\| < \epsilon$, for some $\epsilon > 0$, or even the more sophisticated Data- and Histogram- rules (Lee & Verleysen, 2007), which work with representative prototypes of data instead of input data points.

After an appropriate rule is selected, a weighted graph $G = (V, E)$ is then constructed by using the data and the set of allowed connections. The data are the vertices V, the allowed connections are the edges E, and the edge labels are the Euclidean distances between the corresponding vertices. Sometimes the resulting graph is disconnected and, in order to connect it, some edges must be added using a minimum spanning tree procedure (it must be noted here that not all the similar methods available add edges to connect the resulting graph, as Laplacian Eigenmaps (Belkin & Niyogi, 2003b), for instance, which work with the

connected subgraphs). Finally, the distance matrix of the weighted undirected graph is obtained by repeatedly applying Dijkstra's algorithm (Dijkstra, 1959).

The computed path lengths obtained using the graph construction described above complies with the non-negativity, identity and triangle inequality conditions of a distance function by construction: all the edge labels are greater or equal than zero; the distance between two vertices is zero i.f.f. both vertices are the same; and because of Dijkstra's algorithm. The symmetry condition holds when the graph is undirected, as used in the models defined in this thesis.

Next, we provide some details about the minimum spanning tree procedure used to connect the resulting graph (constructed utilizing a r-rule, where r can be any of the above mentioned rules). In this thesis, we resort to Prim's algorithm (Prim, 1957) in order to find the minimum spanning tree of a given graph, but alternative procedures can be used, as Kruskal's algorithm (Kruskal, 1956), for instance. As a reminder, we summary describe Prim's algorithm.

Algorithm: Prim

Input: A connected weighted graph with vertices V and edges E.
Output: V_{new} and E_{new} describe a minimal spanning tree

Method:

1. $V_{new} = x$, where x is an arbitrary node (starting point) from V, $E_{new} = \{\}$
2. Repeat until $V_{new} = V$:
   (a) Choose edge (u, v) from E with minimal weight such that $u \in V_{new}$ and v is not (if there are multiple edges with the same weight, choose arbitrarily)
   (b) Add v to V_{new}, add (u, v) to E_{new}

We illustrate the procedure with an example that uses the Swiss-Roll dataset described in section 4.4.1. The corresponding constructed graph using the K-rule (which is the most commonly used in graph construction) is shown in Fig. B.1

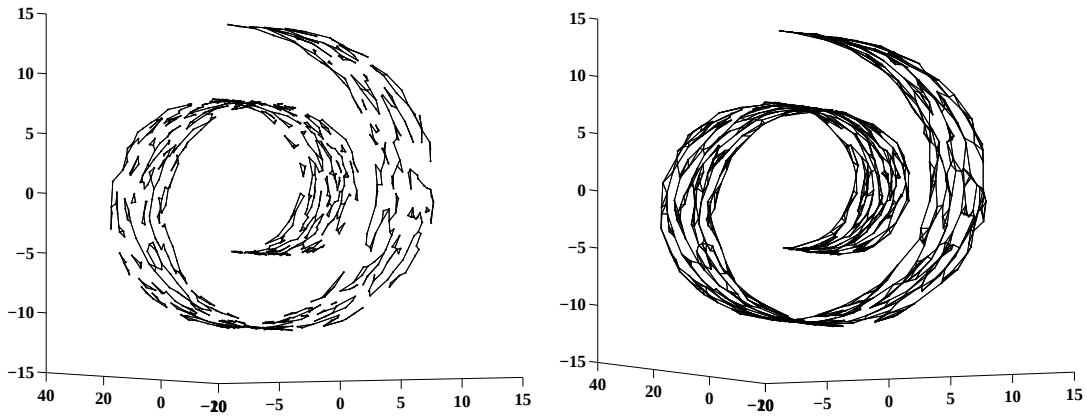

Figure B.1: Graph construction for the *Swiss-Roll* set using the K-rule. (Left): $K = 2$; (right): $K = 4$.

(left), for $K = 2$. Here, we can observe that several connected subgraphs are obtained. However, by setting $K \geq 3$, a connected graph such as the one shown in Fig. B.1 (right, for K=4) is obtained.

When Prim's algorithm is required, some aspects about how the subgraphs are connected should be taken into account. Since we have several connected subgraphs, we use Prim's algorithm to find them. This was accomplished by modifying the step 2.a): when an edge (u, v) cannot be chosen means that we have found a subgraph. It is listed and saved, and Prim's algorithm is run again using only the rest of the vertices. Once all the subgraphs are found, they are linked. For this, two alternatives, of many, can be followed. The first one is linking the first two subgraphs and considering the union of the corresponding vertices as a new component which will be linked with the rest of the subgraphs. The second alternative consists on linking the first two subgraphs and using only the second one to be linked with the rest of the subgraphs.

The connected graphs using the two alternatives for Fig. B.1 (left) are shown in Fig. B.2. It is clear that none of the resulting connected graphs following these alternatives capture the real structure of the dataset. A better and simpler alternative was designed. As in the previous alternatives, the step 2.a) was modified: when the first connected subgraph is found, it is immediately linked with the nearest vertex of the rest of the vertices. This vertex is added to the tree as

in step 2.b), which allows to follow Prim's algorithm as usual. This modification allows to connect the graph and, at the same time, to find the corresponding minimum spanning tree. Figure B.3 shows the connected graph for Fig. B.1 (left) using this alternative. Now, it is observed that the connected graph follows the real structure of the dataset. This last alternative was used in this thesis.

Once a connected graph is obtained, some modifications to the edge weights can be made in order to emphasize some properties of the data and to help to the developed method. For example, in Zhao (2006), a type of similarity that considers both the local geometry information (of both labeled and unlabeled data) and the class information (of labeled data) was used to modify the edge weights of the graph. Another example was presented in Shin *et al.* (2006), where the edge weights are adjusted considering directionality. Note that these edge weights modifications will demand some extra computational time in the entire learning task.

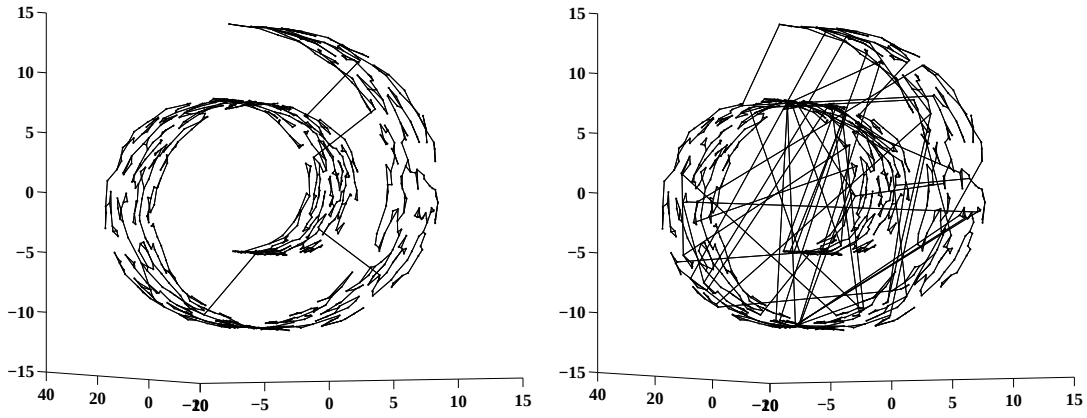

Figure B.2: Connection of subgraphs of Fig. B.1 (left) using the two alternatives described in the main text.

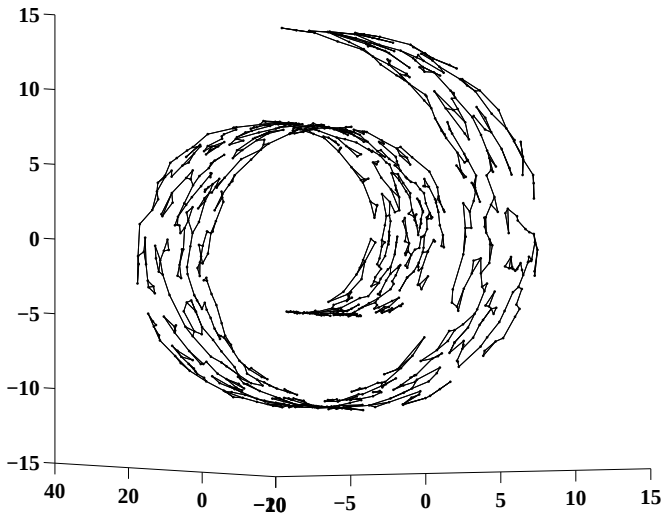

Figure B.3: Connection of subgraphs of Fig. B.1 (left) using the best alternative described in the main text.