UNIVERSITY “JAUME I” OF CASTELLÓN

Superior School of Technologies and Experimental Sciences

DEPARTMENT OF MATHEMATICS

MATHEMATICAL METHODS TO PREDICT THE DYNAMIC SHAPE EVOLUTION OF CANCER GROWTH BASED ON SPATIO-TEMPORAL BAYESIAN AND GEOMETRICAL MODELS

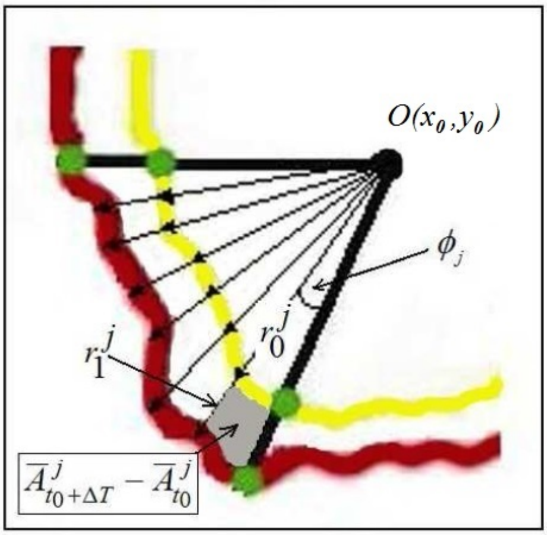
Author:

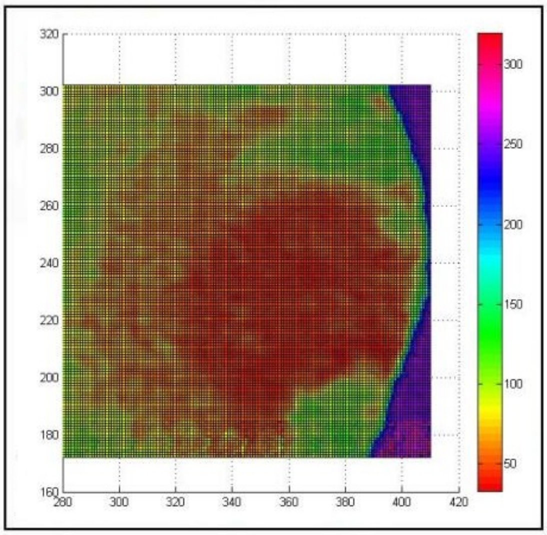
Iulian Teodor Vlad

Advisors:

Prof. Dr. Jorge Mateu

Prof. Dr. José Joaquin Gual Arnau





Castellón de la Plana

2015

University "Jaume I" of Castellón

Superior School of Technologies and Experimental Sciences

DEPARTMENT OF MATHEMATICS

Doctoral Thesis

Mathematical Methods to Predict the Dynamic Shape Evolution of Cancer Growth based on Spatio-Temporal Bayesian and Geometrical Models

---

Author:

Ing. Iulian Teodor Vlad

Supervisors:

Prof. Dr. Jorge Mateu

Prof. Dr. José Joaquin Gual Arnau

A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy
in the
Computer Mathematics

December 2015

Mathematical Methods to Predict the Dynamic Shape Evolution of Cancer Growth based on Spatio-Temporal Bayesian and Geometrical Models

Copyright © 2015 - Ing. Iulian Teodor Vlad
ALL RIGHT RESERVED

Declaration of Authorship

I, Ing. Iulian Teodor Vlad, declare that this thesis entitled, 'Mathematical Methods to Predict the Dynamic Shape Evolution of Cancer Growth based on Spatio-Temporal Bayesian and Geometrical Models' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

DATE: ……………………………………………

SIGNATURE: ……………………………………………

This dissertation is advanced towards to the University "JAUME I" of Castellón from Spain, for fulfilment the requirements in order to degree of **DOCTOR OF PHILOSOPHY ON MATHEMATICAL COMPUTATION** running in the period: oct 2009 - oct 2013, in particular, of fulfill the period of two years (oct 2009 - oct 2011) as well as Formatting Professional Research and another two years (oct 2011 - oct 2013) as Member of the Research Team: Statistical Modeling to Environmental Problems, code 145 coordinated by the Prof. Dr. Jorge Mateu, as requirements of the course 14019 - "Matemática Computacional" on the Superior School of Technologies and Experimental Sciences, Department of Mathematics.

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy:

Director: Name: Prof. Dr. Jorge Mateu

Signature: ....

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy:

Co-Director: Name: Prof. Dr. José Joaquin Gual Arnau

Signature: ....

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy:

Committee Member: Name: ....

Signature: ....

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy:

Committee Member: Name: ....

Signature: ....

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy:

Committee Member: Name: ....

Signature: ....

Motto:

Axiom *5****B***: “***Be*** *good,* ***be****yond* ***be****tter, to* ***be****come the* ***be****st*”.

I.T. Vlad

Abstract

by Ing. Iulian Teodor Vlad

The aim of this research is to observe the dynamics of cancer tumors and to develop and implement new methods and algorithms for prediction of tumor growth. We offer some tools to help physicians for a better understanding and treatment of this disease. Using a prediction method, and comparing with the real evolution a physician can note if the prescribed treatment has the desired effect, and according to this, if necessary, to take the decision of surgical intervention.

In this thesis we analyze the spatio-temporal dynamics of shape evolution and we apply these to a particular case of brain tumors.

The plan of the thesis is the following. In Chapter 1 we briefly recall some properties and classification of points processes with some examples of spatio-temporal point processes. Chapter 2 presents a short overview of the theory of Lévy bases and integration with respect to such basis is given, we recall standard results about spatial Cox processes, and finally we propose different types of growth models and a new algorithm, the Cobweb, which is presented and developed based on the proposed methodology. Chapters 3, 4 and 5 are dedicated to present new prediction methods. The implementation in Matlab software comes in Chapter 6. The thesis ends with some conclusion and future research.

Acknowledgements

Special thanks to my advisor Prof. Dr. Jorge Mateu for his collaboration and to University “JAUME I” of Castellón for the inestimable help, supporting and invaluable experience offered, an educational institution of study nearby of citizens and nature, with a great human and material basis, designed to provide a permanently support to the current development of actual society.

Also we want to thank to our colleague Francisco, who offer us the right to use the image acquisitions of the tumor images based on which we have shown the functionality of these algorithms.

Contents

Title ii

Copyright iii

Declaration v

Approve vii

Motto ix

Abstract xi

Acknowledgements xiii

Table of Contents xv

List of Figures xix

List of Tables xxi

Abbreviations xxiii

Notations xxv

Dedication xxvii

1 Introduction: basics of stochastic processes 1
- 1.1 Temporal processes 12
- 1.2 Spatial processes 14
  - 1.2.1 Types of spatial processes 15
- 1.3 Point processes 15
- 1.4 Spatial point processes 27
  - 1.4.1 Regular processes 30
  - 1.4.2 Cluster processes 31
  - 1.4.3 Complete spatial randomness 32
- 1.5 Marked spatial point processes 33
- 1.6 Poisson processes 35
  - 1.6.1 Homogeneous Poisson process 39

1.6.2 Non-homogeneous (inhomogeneous) Poisson process 39
1.6.3 Spatial Poisson processes 40
1.7 Cluster processes 41
1.8 Spatio-temporal point processes 46
1.8.1 Earthquake processes 48
1.8.2 Explosion processes 49
1.8.3 Birth-death processes 49
1.8.4 Point patterns sampled in time 50
1.8.5 Intensity measures of spatio-temporal point patterns (First-Order Properties) 51
1.8.6 Second-order intensities 54

2 Spatial point processes for tumor growth. Cobweb algorithm[1] 57
2.1 Spatial Cox point processes 60
2.1.1 Lévy-based Cox processes 61
2.1.2 Lévy-based tumor growth modeling 63
2.2 Modeling tumor growth: a new algorithm 69
2.3 Software 76
2.3.1 Input data 76
2.3.2 Procedures that must be fulfilled 76
2.4 Real data analysis 78
2.5 Conclusions 81

3 Geometric prediction methods of tumor growth[2] 83
3.1 Methodology 86
3.1.1 Shape and growth description 86
3.1.2 Normal method 87
3.1.2.1 Curve evolutions 93
3.1.3 Radius method 94
3.2 Simulations and application 94
3.2.1 Simulated data: random curves 94
3.2.2 Simulated data: parametric curves 97
3.2.3 Real data 98

4 Bayesian prediction of tumor growth[3] 101
4.1 Data set 103
4.1.1 Image registration 104
4.1.2 Preparing the data 106
4.2 Methodology 109
4.2.1 Statistical framework 109
4.2.2 Statistical inference 110

---

[1] This chapter is based on the published paper: "*A geometric approach to cancer growth prediction based on Cox Processes*" by Vlad and Mateu (2014)[1]

[2] This chapter is based on the published paper: "*Two handy geometric prediction methods of cancer growth*" by Vlad et al. (2015)[2]

[3] This chapter is based on the published paper: "*Bayesian spatio-temporal prediction of cancer dynamics*" by Vlad et al. (2015)[3]

4.2.2.1 SPDE approach . . . 110
4.2.2.2 Bayesian computation . . . 115
4.3 Modeling results . . . 119

5 Functional prediction of tumor growth[4] 125
5.1 Parametric contour functions of Gioblastoma Multiform: the problem of registration by FDA . . . 126
5.2 Principal differential analysis and tumor growth . . . 129
5.3 Application: a Gioblastoma Multiform study . . . 132
5.3.1 Brain scans processing: from images to contour functions . . 133
5.4 Results of principal differential analysis on the contour functions . . 135

6 Software: Prediction of the Dynamic Shape Evolution of Cancer 139
6.1 Image processing . . . 144
6.2 Tumor contour . . . 145
6.3 Prediction of the dynamic shape evolution using the Cobweb algorithm . . . 146
6.4 Bayesian prediction . . . 148
6.5 Geometrical prediction . . . 150
6.6 Logical prediction in space and time . . . 151

7 Conclusions and future research 153

A List of Matlab functions 159

References 165

---

[4] This chapter is based on the submitted paper: "*Principal differential analysis for modeling dynamic contour evolution. A distance-based approach for the analysis of Gioblastoma Multiform*" by Romano et al. (2014)[4]

List of Figures

1.1 Several distributions of points in a 2-dimensional region: a) Cluster distribution; b) Poisson distribution; c) Regular distribution 12
1.2 A sequential inhibition processes (Diggle, 1976) 30
1.3 A Poisson cluster process 31
1.4 A CSR pattern 32
1.5 An earthquake process 48
1.6 An explosion process 49
1.7 A birth-death process 50
1.8 Points patterns sample in time 50

2.1 The star-shape object Y_t. 64
2.2 Stochastic representation of $\bar{A}_t(\phi)$. 66
2.3 Brain tumor (a) and its location (b) 69
2.4 Second image acquisition after one month: a) original image acquisition; b) boundary of tumor (red) within the sample space (blue) . 70
2.5 Superimposed images of tumor 71
2.6 Tumor growth in star-shape 71
2.7 Calculation of the growth tumor at time $t + \Delta T$ 73
2.8 Error propagation 75
2.9 Predicted tumor after two months 75
2.10 Diagram of the script 78
2.11 Real data analysis. A yellow line represents the tumor at time t (time when it was discovered), and a red line represents the tumor after time $t + \Delta T$. 80
2.12 Predicted tumor 81

3.1 Signed curvature κ for a negatively oriented planar closed curve α . 88
3.2 Evolution in time of a tumor brain cancer: a) first image acquisition of curve α_1 and a points $P_i^{t_1}$; b) the same tumor after time Δt . . . 90
3.3 Calculus of the predicted points $\overline{P_k^{t_3}}$ with the normal method 91
3.4 Simulated tumors at time t, $t+\Delta t$ and $t+2\Delta t$ using random curves: a) prediction with radius method; b) prediction with normal method 95
3.5 Parametric and simulation shapes of curve at time t_1, t_2 and t_3: a) prediction with radius method; b) prediction with normal method . 97
3.6 Evolution in time of the real brain tumor: a) boundary of the tumor at time t; b) boundary of the same tumor in december; c) boundary of the tumor in January 99

3.7 Brain tumor: real evolution vs. prediction: a) prediction with radius method; b) prediction with normal method 100

4.1 Original CT image acquisition 103
4.2 Registered and normalized images 105
4.3 Image histogram and thresholding 107
4.4 ROI in registered images (left) and overlapped points for infected cells in ROI (right) 109
4.5 Correlations between the predicted and observed data for all models. From top left to bottom right: Model $M1$ for November, model $M1$ for December, model $M1$ for January, model $M2$ for November, December, January, model $M3$ (bottom left), model $M4$ (bottom middle) and the complete model $M5$ with covariable distance (bottom right). 122
4.6 Posterior distribution of parameters for model M5: $Intercept = 2.7470$ (top left); $\phi = 2.68945162$ (top right); $Normal\,Variance(\sigma_X^2) = 0.5815021$ (bottom left); $Practical\ Range = 107.1021$ (bottom right).123
4.7 Prediction map based on model $M5$ 123

5.1 Contour function. The red line identifies the first step of observation, the green one the second step 128
5.2 Tumor boundary from different patients at different times. 134
5.3 Registered curves: $X_i(s), i = 1, \ldots, 15$ (top left); $Y_i(s), i = 1, \ldots, 15$ (top right) $X_i^*(s), i = 1, \ldots, 15$(bottom left) $dY_i^*(s), i = 1, \ldots, 15$(bottom right) 135
5.4 Functional boxplot for the first step. 135
5.5 Residual function 136
5.6 Estimated coefficient functions for the X component. From top left to bottom right: $\alpha_{x*}(s), \beta_{x*1}(s)$ for the first step; $\alpha_x(s), \beta_{x1}(s)$ for the second step. 137
5.7 Estimated coefficient functions $\beta_{2x}(s), \beta_{2x*}(s)$ for the first and second step 137

6.1 Diagram of PreDySEC 141
6.2 The main interface of PreDySEC 142
6.3 Image processing module 144
6.4 Automatic tumor boundary 145
6.5 The prediction with Cobweb module 147
6.6 Bayesian prediction interface 149
6.7 Geometrical predictions 151
6.8 Logical prediction module 152

List of Tables

3.1 Absolute and relative errors for areas obtained from the normal and radius methods. 96

4.1 DIC, CPO and nEp for each model 119
4.2 Fixed effects: Intercept 121
4.3 Fixed effects: Distance 121
4.4 Correlation coefficients for each model 121

5.1 Models fitting, R^2 goodness of fit. 136

Abbreviations

ACF	Auto-Correlation Function
ARMA	Auto-Regressive Moving Average
CPO	Conditional Predictive Ordinate
CSMS	Complete Separable Metric Space
CSR	Complete Spatial Random
CT	Computer Tomography
DIC	Deviance Information Criterion
FDA	Functional Data Analysis
GBM	Blioblastoma Multiforme
GF	Gaussian Field
GMRF	Gaussian Markov Random Field
GUI	Graphical User Interface
IGRT	Image Guided Radiation Therapy
IMRT	Intensity Modulated Radiotherapy
INLA	Integrated Nested Laplace Approximation
LGCP	Log-Gaussian Cox Processes
MCMC	Markov Chain Monte Carlo
MRI	Magnetic Resonance Images
PDA	Principal Differential Analysis
PDE	Principal Differential Equation
PDF	Probability Density Function
PreDySEC	Prediction of the Dynamic Shape Evolution of Cancer
RMSE	Root Mean Square Error
ROI	Region Of Interest
RV	Random Variable
RW1	Random Walk model of order 1
SPDE	Stochastic Partial Differential Equations

Notations

In this work, we use the following notations throughout the text:

$\mathbb{R}$	real line
$\mathbb{R}^d$	dimensional space
$\mathbb{R}_+$	nonnegative real numbers
$\mathbb{Z}, \mathbb{Z}_+$	integers of $\mathbb{R}, \mathbb{R}_+$
X	random variable
Ω	space of probability elements ω
$(\Omega, \mathcal{F})$	measurable space
$P(A)$	probability measure
$(\Omega, \mathcal{F}, P)$	probability space
ε	measurable sets in probability space
(Ω, ε, P)	basic probability space
$\mathcal{S}$	sample space
W	observable space
$\emptyset, \emptyset(\cdot)$	null set, null measure
$A \subseteq \Omega$	subset of Ω
A^n	n-fold product set $A \times \cdots \times A$
$\mathcal{A}$	family of sets generating $\mathcal{B}$
$\mathcal{B}$	Borel σ-field
$\mathcal{B}_b(\Omega)$	bounded Borel subset of Ω
$E(X)$	expectation or mean or average value
$E(X\|\mathcal{A})$	conditional expectation
$f(x)$	probability density function
$F_X(x)$	distribution function
$N(A)$	number of points in set A
$N(a,b]$	number of points in interval $(a,b]$
$N(t)$	$N(t) = N(0,t] = N\big((0,t]\big)$
$N(\cdot)$	point process
$N_m(\cdot)$	marked point process
$N_c(\cdot)$	cluster process

$G[h]$	probability generating functional (p.g.fl.) of N
$G[h\|x]$	member of measurable family of p.g.fl.'s
G, G_I	expected information gain of N
$K(r)$	Ripley K-function
t_i	instance of time
ΔT	time interval
T_i	forward recurrence time
$r = l(r)$	$=\parallel s_i - s_j \parallel$ distance between s_i and s_j
$\lambda(\cdot)$	intensity function of a point process
$\lambda_2(\cdot,\cdot)$	second order-intensity function
$V(A)$	$= \mathrm{Var} N(A)$variance function of N or ξ in $\mathbb{R}$
$c(k)$	$= \mathrm{Cov}(X_t, X_{t-k})$ covariance function
$C(\cdot)$	$C_{[k]}(\cdot)$ factorial cumulant measure and density
$L(A)$	Gaussian Lévy-basis
$k(\cdot,\cdot)$	probability kernel
$P(x, A)$	Markov transition kernel
$\kappa_j(\cdot)$	the jth cumulant moment of the spot variable
$\Lambda(\xi)$	intensity field
$\ell(\cdot)$	Lebesgue measure in $\mathcal{B}(\mathbb{R}^d)$

The rest of notation used in this dissertation is explained as it appears in the text.

Dedicated to my family, especially to my mother who did not live to see the results of this research.

Chapter 1

Introduction: basics of stochastic processes

Cancer is a widely spread disease that affects a large proportion of the human population. Recently developed technologies such as controlled chemotherapy, IMRT (intensity modulated radiotherapy), IGRT (image guided radiation therapy) and hadronotherapy, do provide good results in the detection, control and follow-up of cancer growing. In this context, it is needed a precisely detection of the tumor boundary, and a further *prediction* of the tumor dynamics to verify the result of a particular treatment.

Prediction, a statement about the *future*, is meant to have a a priori information about one event that can be observable. Over time, this subject has created a lot of controversy and discussion. From ancient times to present, people have tried to find and predict some facts and how they evolve in the *future*. The power to have such information sometimes turns on the perception that this is magic (e.g. to know when a solar eclipse will happen in the Maya civilisation, to know when the wind will turn on the direction in a sea battle of ships in 17th-18th Century, to depict the cholera source by clusters in London 1854, etc.). This kind of scientific worries was contemned but a number of researchers were continuing to study and make experiments to develop revolutionary theories and methodologies for prediction.

Talking about *future* it is obvious that we must refer to time and to the next values of this abstract variable. Actually, even mathematical formalism can contain multiple abstract notions may be hard to explain and analyze intuitively. For example the “lineal element” defined by eq.(3) in the article “The foundation of general relative theory of relativity” by Albert Einstein, has no direct physical meaning.

In this research we are concerned with observing and modeling the effect of time on the evolution of cancer.

In modern societies, mathematicians and statisticians developed theories and methodologies to equipe with some probability the realization of one event in the study experiment. Some evolution aspects over time and the dynamic of phenomenon can be defined a priori if we have enough input data and the developed theory to do this. Otherwise we can use empirical methods,based on input data and the researcher experience, to find an approximate solution of the interest variable.

If we look at the pairs of terms synthetic/analytic and a priori/a posteriori we see that a mathematical interpretation can be represented diagrammatically as follows: the overlap of the synthetic with the a priori indicates that he asserts that there are synthetics statements a priori, but we would also regard logical laws and certain fundamental principles of mathematic as a priori synthetic. By means of these distinctions between the a priori and a posteriori, and between the analytic and synthetic, is called formalism. Also the formalism can be defined in simple words as a description of something in formal mathematical or logical terms.

Mathematical formalism always tries to find a compromise between simplicity of analysis and requirements of realism. On the one hand, we have extremely complex natural and biological systems; on the other hand, we need to formally address some quantitative issues about these systems which can be often done only through the use of mathematical models that may rest on grossly over-simplified assumptions.

On some occasions, a particular mathematical formalism seems to be "pre-adapted" to a variety of natural and biological systems and can be profitably used to model a diverse set of processes.

Stochastic Geometry, Bayesian methods, Inference Methods, SPDE, Functional data Analysis are some class of such models, used now to solve real problems in the field of:

- epidemiology (home locations of infected patients),
- computational neuroscience (spikes of neurons),
- forestry and plant ecology (positions of trees or plants in general),
- meteorology (weather prediction),
- geography (positions of human settlements, towns or cities),
- seismology (epicenters of earthquakes),
- materials science (positions of defects in industrial materials),
- astronomy (locations of stars or galaxies, revealing regularity in the spatial distribution of point-like objects, identification of important scales in the spatial distribution of point-like objects, etc.),
- stellar statistics (deriving distributions, testing of predicted distribution functions, identification of clusters and associations of stars, search for wide binaries and multiple systems),
- cosmological problems (testing of predicted distribution functions, identification of galaxy clusters, voids, etc.),
- medicine (snapshots of a growing brain tumors),
- zoology (burrows or nests of animals),
- communication systems (wireless network, telephony), etc.

In this research we analyze the spatio-temporal dynamics of shape evolution in order to develop new prediction methods and we apply these to a particular case of brain tumors. These objects are originally processed from CT and MRI images, and can be depicted as a collection of image pixels with varying degrees of color intensity levels. We consider spatio-temporal stochastic processes within a Bayesian framework to model spatial heterogenity, temporal dependence and spatio-temporal interactions amongst the pixels, providing a general modeling framework for such dynamics. We aim at predicting cancer growth in space and time. We analyze real data on brain tumor based on a set of images taken at several visits and also simulated closed curves to randomly generate the tumor cancer contours.

Let us begin with a brief introduction to stochastic theory, define the basic notations and show some important characteristics and types of point processes.

A *stochastic process*, or sometimes a *random process*, is the counterpart to a deterministic process (or deterministic system) in probability theory. Instead of dealing with only one possible“reality” of how the process might evolve under time (as is the case, for example, for solutions of an ordinary differential equation), in a stochastic or random process there exist some indetermination in its future evolution described by probability distributions. This means that even if the initial condition (or starting point) is known, there are many possibilities when the process might go, but some paths are more probable and others less (Cox (1994)[5] and Daley and Vere-Jones (2003)[6]).

In the simplest possible case (“discrete time”), a stochastic process amounts to a sequence of random variables known as a time series (for example, see Markov chain). Another basic type of a stochastic process is a random field, whose domain is a region of space, in other words, a random function whose arguments are drawn from a range of continuously changing values. One approach to stochastic processes treats them as functions of one or several deterministic arguments ("inputs", in most cases regarded as "time") whose values ("outputs") are random variables: non-deterministic (single) quantities which have certain probability distributions. Random variables corresponding to various times (or points, in the case of random fields) may be completely different. The main requirement is that these different random quantities all have the same "type". Although the random values of a stochastic process at different times may be independent random variables, in most commonly considered situations they exhibit complicated statistical correlations.

When we use the word *"point"* we shall refer to an object or an event in a *"location"* or in the *"sample space"*.

The sample space will be denoted by $\Omega \subseteq \mathcal{S} \subset \mathbb{R}^d$ dimensional space (usually the Euclidean space, with d=2 or d=3 in applications).

The distribution of such n points in the sample space or in the *"study region"* Ω can be random or governed by some laws. Each point n must accomplish at least one condition to be observed (it represents a value of the observed phenomenon) and can contain also some supplementary *"information"*.

If we talk about *"point process"* in terms of stochastic theory, then we shall deal with words like: *"random"* or *"probability"* refer to values, and *"measure"* or *"topology"* with reference to the sample space.

Let us consider that we have n points, in an observable space Ω included or equal in the sample space $\mathcal{S}$ ($\Omega \subseteq \mathcal{S}$) and let A be a realization of these events.

Then:

$$A = \{\omega_1, \omega_2, \dots, \omega_n\}.$$

Definition 1.1. The **"atoms"** of Ω are the events $\{\omega\}$ having just one outcome $\omega \in \Omega$.

We say that the atom $\{\omega\}$ of the outcome ω is the event ***"ω occurs"***

The events are subsets of the sample space and by additivity, any probability measure P must satisfy:

$$P(A) = P(\{\omega_1\}) + P(\{\omega_2\}) + \dots + P(\{\omega_n\}) \tag{1.1}$$

Every probability measure P on a finite sample space Ω is determined by its values on the atoms. The value on an arbitrary event $A \subseteq \Omega$ is then computed by the formula:

$$P(A) = \sum_{\omega \in A} P(\{\omega\}) \tag{1.2}$$

The values of P on the atoms may be assigned arbitrary as long as:

1. For every atom $\{\omega\}$, $0 \leq P(\{\omega\}) \leq 1$
2. $\sum_{\omega \in \Omega} P(\{\omega\}) = 1$

whenever (1) and (2) hold, P defines a ***consistent probability measure*** on Ω.

Definition 1.2. Let $\Omega \neq \emptyset$. A **field** on Ω is a system $\mathcal{A}$ of subsets $A \subseteq \Omega$ satisfying the following conditions:

1. $\Omega \in \mathcal{A}$, $\emptyset \in \mathcal{A}$
2. If $A_1, A_2 \in \mathcal{A}$ then $A_1 \cup A_2 \in \mathcal{A}$ and $A_1 \cap A_2 \in \mathcal{A}$
3. If $A \in \mathcal{A}$ then $A^c \in \mathcal{A}$

Definition 1.3. A **content** is a set function μ defined on a field $\mathcal{A}$ such that:

1. $\mu(A) \in [0, \infty]$ whenever $A \in \mathcal{A}$
2. $\mu(\emptyset) = 0$
3. $\mu(A_1 \cup A_2) = \mu(A_1) + \mu(A_2)$ whenever $A_1, A_2 \in \mathcal{A}$ and $A_1 \cap A_2 = \emptyset$

Definition 1.4. Let $\mathcal{A}$ be a field and let $\mu|\mathcal{A}$ be a content. The content μ is called **σ-additive** if:

$$\mu\left(\bigcup_{i \in \mathbb{N}} A_i\right) = \sum_{i \in \mathbb{N}} \mu(A_i) \tag{1.3}$$

for every pairwise disjoint sequence $(A_i)_{k \in \mathbb{N}} \subseteq \mathcal{A}$ such that: $\bigcup_{i \in \mathbb{N}} A_i \in \mathcal{A}$.

If a content is σ-additive then the content has several continuity properties which make calculations easier.

Definition 1.5. A field $\mathcal{F}$ on Ω is a σ-field if:

$$(A_i)_{i \in \mathbb{N}} \subseteq \mathcal{F} \Rightarrow \bigcup_{i \in \mathbb{N}} A_i \in \mathcal{F} \tag{1.4}$$

A pair $(\Omega, \mathcal{F})$ where $\mathcal{F}$ is a σ-field on Ω is called a **measurable space**.

Definition 1.6. A σ-additive content which is defined on a σ-field is called a **measure**.

A probability space is a measure space such that the measure of the whole space is equal to 1.

Definition 1.7. A **probability space** is a triplet $(\Omega, \mathcal{F}, P)$ consisting of a set Ω (called the sample space), a σ-algebra (also called σ-field) $\mathcal{F}$ of subsets of Ω (these subsets are called events), and a measure P such that $P(\Omega) = 1$ (called the probability measure).

Definition 1.8. Let $(\Omega, \mathcal{F}, P)$ be a probability space and let $\mathcal{A} \subseteq \mathcal{F}$ be a sub-σ-field. Let X be a nonnegative or integrable random variable. The **conditional expectation** $E(X|\mathcal{A})$ of X given $\mathcal{A}$ is an $\mathcal{A}$-measurable random variable Y satisfying:

$$\int_A X\,dP = \int_A Y\,dP, \text{ for all } \ A \in \mathcal{A} \tag{1.5}$$

Definition 1.9. Let $(\Omega, \mathcal{F}, P)$ be a probability space. Any $\mathcal{F}$-measurable real-valued function $X : \Omega \to \mathbb{R}$ is called a **random variable (R.V.)**.

An integer random variable is a function X defined on a sample space Ω, that takes only integer values. Namely, for every sample point $\omega \in \Omega$, $X(\omega)$ is a integer. The probability distribution of X is the sequence of numbers p_n such that p_n is the probability of the event "X equals n". The event "X equals n" can be writen $(X = n)$. As a subset of Ω this event is:

$$(X = n) = \{\omega \in \Omega : X(\omega) = n\}$$

Definition 1.10. Suppose X is a integer R.V. with distribution $p_n = P(X = n)$. The **expectation** or **mean** or **average value** of X is:

$$E(X) = \sum_n n \cdot p_n = \sum_n n \cdot P(X = n) \tag{1.6}$$

Definition 1.11. Let X be a random variable. Then the function $F_X : \mathbb{R} \to [0, 1]$ is the **distribution function** of X defined by:

$$F_X(x) := P(X \leq x), x \in \mathbb{R} \tag{1.7}$$

and satisfies:

1. $F(-\infty) = 0$
2. $F(\infty) = 1$

3. $F(x) \leq F(x')$ if $x < x'$.

The distribution of X is P_X, i.e. the image of P under X defined by:

$$P_X(B) := P(X^{-1}(B)) = P(X \in B), B \in \mathcal{B},$$

where $\mathcal{B}$ is the Borel σ-algebra.

Thus, the distribution function F_X determines the values of the distribution P_X on intervals by:

$$P_X((a,b]) = P\left(\{a \leq X \leq b\}\right) = F(b) - F(a).$$

A probability distribution has density *f*.

Definition 1.12. Let X be a random variable. Then the function $f : \mathbb{R} \to \mathbb{R}$ is the **probability density function (P.D.F.)** of the random variable X if:

$$P_X((a,b]) = P\left(\{a \leq X \leq b\}\right) = \int_a^b f(x)dx \tag{1.8}$$

and satisfies:

1. $F(x) = P(\{X < x\}) = \int_{-\infty}^{x} f(x)dx.$
2. $f(x) > 0,\ \forall x \in \mathbb{R}$
3. $\int_{-\infty}^{\infty} f(x)dx = 1$
4. *if* $X \in (a,b)$, *then* $f(x) = 0$ *for* $\forall x \notin (a,b)$
5. $\int_a^b f(x)dx = F(b) - F(a)$

Example 1: Consider the experiment of flipping a coin once. Then:

$$\Omega = \{H, T\} \text{ (the possible outcomes: “Heads” and “Tails”)}$$
$$\mathcal{F} = P(\Omega)\ (\mathcal{F} \text{ contains all subsets of } \Omega)$$
$$P(\{H\}) = P(\{T\}) = \frac{1}{2}$$

Example 2: We pick a real number at random in the interval [0, 2]. $\Omega = [0,\ 2]$, $\mathcal{F}$ is the Borel σ-field of [0, 2]. The probability of an interval $[a, b] \subset [0, 2]$ is:

$$P([a,b]) = \frac{b-a}{2}$$

Example 3: If A is an event in a probability space, the random variable:

$$1_A(\omega) = \begin{cases} 1, & \omega \in A; \\ 0, & \omega \notin A. \end{cases}$$

is called the indicator function of A. Its probability law is called the ***Bernoulli distribution*** with parameter $p = P(A)$.

The set of all possible sequences is called "the Bernoulli sample space" Ω, and the correspondent experiment process is called "the Bernoulli process"

Example 4: We say that a random variable X has the ***Binomial law*** $B(n, p)$ if:

$$P(X = k) = \begin{pmatrix} n \\ k \end{pmatrix} p^k (1-p)^{n-k} \tag{1.9}$$

for $k = 0, 1, 2, \ldots, n$

Example 5: We say that a random variable X has the ***Normal law*** $N(m, \sigma^2)$ if:

$$P(a < X < b) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_a^b e^{-\frac{(x-m)^2}{2\sigma^2}} dx \tag{1.10}$$

Example 6: If X is a random variable with normal law $N(0, \sigma^2)$ and λ is a real number,

$$\begin{aligned} E(\exp(\lambda X)) &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{\lambda x} e^{-\frac{x^2}{2\sigma^2}} dx \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{\sigma^2\lambda^2}{2}} \int_{-\infty}^{\infty} e^{-\frac{(x-\sigma^2\lambda)^2}{2\sigma^2}} dx \qquad (1.11) \\ &= e^{\frac{\sigma^2\lambda^2}{2}} \end{aligned}$$

Example 7: Consider an experiment that consists of counting the number of traffic accidents at a given intersection during a specified time interval.

$$\begin{aligned} \Omega &= \{0, 1, 2, \ldots\} \\ \mathcal{F} &= P(\Omega) \ (\mathcal{F} \text{ contains all subsets of } \Omega) \\ P(\{k\}) &= e^{-\lambda} \frac{\lambda^k}{k!} \ (\textbf{\textit{Poisson probability}} \text{ with parameter } \lambda > 0) \end{aligned}$$

Example 8: If X is a random variable with Poisson distribution of parameter $\lambda > 0$, then:

$$E(X) = \sum_{n=0}^{\infty} n \frac{e^{-\lambda}\lambda^n}{n!} = \lambda e^{-\lambda} \sum_{n=1}^{\infty} \frac{e^{-\lambda}\lambda^{n-1}}{(n-1)!} = \lambda$$

With these concepts in mind we can now state that:

- if in the probability space $(\Omega, \mathcal{F}, P)$, the random variable X has associated a time series then we have a ***"temporal process"***
- if the sample space is d dimensional ($d \geq 2$) Euclidean then we have a ***"spatial process"***
- if in the probability space $(\Omega, \mathcal{F}, P)$, the measure $N(A)$ represents the number of points falling in the subset A of $\Omega \subseteq \mathcal{S}$, then we have a ***"point process"***

- if in the probability space $(\Omega, \mathcal{F}, P)$, each event occurred in the subset A of $\Omega \subseteq \mathcal{S}$ has associated a location $\mathbf{s}$ then the measure $N(s)$ represents the number of events that are considered and we have a ***"spatial point process"***
- if the point process $N(s)$ is *"marked"* then we have a particular point process named ***"marked point process"***
- if in the case of point process $N(s)$, the random variable X has a Poisson distribution then we have a ***"poisson point process"***
- if in the case of point processes $N(s)$ the random variable X in distributed in clusters over the sample space then we have a ***"cluster point process"***
- if the time series values t_i are index the spatial random set $\Omega \subseteq \mathcal{S}$ of spatial point processes $N(\boldsymbol{s})$ then we have a ***"spatio-temporal point process"*** $N(\boldsymbol{s}, t)$

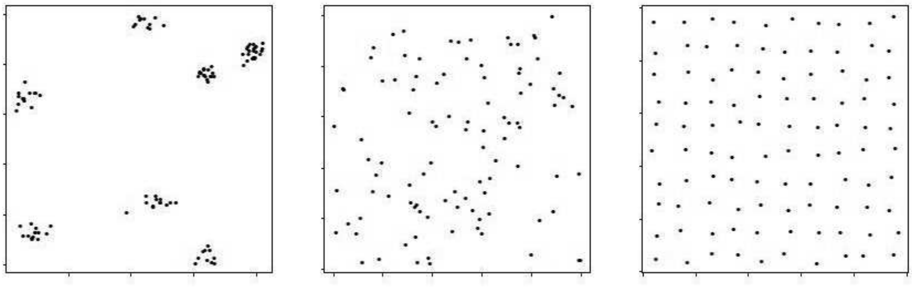


Figure 1.1: Several distributions of points in a 2-dimensional region: a) Cluster distribution; b) Poisson distribution; c) Regular distribution

In Figure 1.1 it is shown several possible distributions of points in a region. They are realizations of spatial point processes.

1.1 Temporal processes

Definition 1.13. A stochastic process observed over time is called a **time series**.

Define a real-valued time series by:

$$\{X_t \in \mathbb{R} : -\infty < t < \infty\} \tag{1.12}$$

the stochastic process is said to be a second-order (weakly) stationary if:

$$E(X_t) = \mu(= 0) \tag{1.13}$$

and

$$\text{Cov}(X_t, X_{t-k}) = c(k) \tag{1.14}$$

where $c(k)$ is the auto-covariance function and is completely specified by the time lag k.

Analysis of time series data is typically conducted in the time or in the frequency domain. The analysis of the time series autocorrelation function (ACF) is discussed in Box et al. (1994)[7], Bhansali (1980)[8], Brockwell and Davis (1991)[9], Hamilton (1994)[10], etc. Estimating, for example, Auto-Regressive Moving Average (ARMA) models is straightforward in the time domain (Postcher and Srinivasan (1994)[11]).

Fuller (1976)[12], Harvey (1981)[13], Lűtkepohl (1991)[14], etc, prefer the time domain approach because of the relative ease of interpretation. Conversely, the frequency domain of time series data is steeped in the Hilbert space algebra. However, the spectral analysis has advantages in its nonparametric approach to data analysis. Although ARMA models can be obtained in the frequency domain, the spectral approach does not require any parametric model for inference.

1.2 Spatial processes

Definition 1.14. A stochastic process with a spatial domain is called a **spatial process**. A spatial process is defined by:

$$\{N(\mathbf{s}) : \mathbf{s} \subseteq \Omega \subset \mathbb{R}^d\} \tag{1.15}$$

where Ω is an index set and $\mathrm{N}(\boldsymbol{s})$ is the attribute of interest at location $\boldsymbol{s}$.

For simplicity, the dimension of space is $\mathbb{R}^2$, representing observations in the plane. The main difference between (1.12) and (1.15) is that the *"information"* about the locations is not necessary well-ordered like a temporal process which is directed in time ($t = \{t_1, t_2, t_3, \ldots , t_n\}$, where $t_1 < t_2 < t_3 < \cdots < t_n$). Time flows unidirectionally, whereas there is no equivalent to past, present, or future in a spatial domain. For this reason many of the methods used to analyze time series must be modified to be appropriate in the spatial context and many techniques of spatial data analysis have been developed independently to time series analysis.

In the geostatistical literature $N(\mathbf{s})$, the attribute of interest observed at location $\mathbf{s}$, is often viewed in the context of random functions (see, e.g., Juornel and Huijbregts (1978)[15], Goovaerts (1997)[16], Chilés and Delfiner (1999)[17]).

Briefly, $N(\mathbf{s}, \omega)$ depends on the realization ω of a random experiment. For a given realization $N(\cdot, \omega)$ is a function of spatial locations and n observations $N(\mathbf{s}_1), N(\mathbf{s}_2), ..., N(\mathbf{s}_n)$ represent an n-dimensional sample of size one from the set of all possible random functions. The stochastic behavior of the attribute N at location $\mathbf{s}$ is induced by considering all possible realizations of the random experiment at that location $N(\mathbf{s})$.

1.2.1 Types of spatial processes

Viewing spatial processes as stochastic processes according to (1.15) is general in that the nature of the index set Ω permits the definition of different types of spatial data. Three spatial data types are defined as follows, according to Cressie (1993)[18]:

I. Geostatistical Data: $N(\mathbf{s})$ is a random variable observed at locations $\mathbf{s} \in \Omega$, where Ω is **fixed and continuous**.

Examples: Random or systematic sampling of a surface, plant yields across a corn field, drilling for ore.

II. Lattice Data: $N(\mathbf{s})$ is a random variable observed at locations $\mathbf{s} \in \Omega$, where Ω is **fixed and discrete**.

Examples: Unemployment rates by census tracts, coloring on remotely sensed pixel images.

III. Point Patterns: $N(\mathbf{s})$ is a random variable observed at locations $\mathbf{s} \in \Omega$, where Ω is a **random set of indices**.

Examples: Positions of lunar craters, locations of trees in a forest, residences that reported break-ins in 1999.

1.3 Point processes

The theory of point processes has undergone an explosive expansion in the last two decades. Point processes and random measures are common in many physical applications found in engineering, astrology, biology, etc. These processes can be observed in one-dimension as a time series or two-dimensions as a spatial point pattern with extensive amounts of literature devoted to their analysis.

The analysis of point pattern data in a compact subset $\mathcal{S}$ of R^n is a major object of study within spatial statistics. There are different ways to build and characterize a point process (using finite-dimensional distributions, void probabilities, capacity functionals, or generating functions). An easier way to build a point process is by transforming an existing point process (by thinning, superposition, or clustering).

Point processes are covered in detail by Bartlett (1975)[19] and Daley and Vere-Jones (2003)[6]. Bartlett (1964)[20], [21], Bartlett (1975)[19] extends analysis of the spatial point process to the frequency domain through the spatial periodic functions.

Ripley (1976)[22] introduced the analysis in the spatial domain through K-function. Currently, Ripley's approach to studying the dependency structure of point patterns remains the dominant method for analysis. Daley and Vere-Jones (2003)[6] in their book: *An Introduction to the Theory of Points Processes*, offer a complete background about this type of Stochastic Processes.

Informally, a point process on a suitable state space $\mathcal{S}$ is understood to be a locally finite collection of distinct random elements $k, (k = 1, 2, ..., n)$ in $\mathcal{S}$. With these specifications we can make the first general definition:

Definition 1.15. A ***point process*** is a random distribution of points in a sample space.

In mathematics, a point process is a random element whose values are "point patterns" on a set $\mathcal{S} \subseteq \mathbb{R}^d$. While in the exact mathematical definition a point pattern is specified as a locally finite counting measure, it is sufficient for more applied purposes to think of a point pattern as a countable subset of $\mathcal{S}$ that has no limit points.

Definition 1.16. A point process N is a stochastic model governing the locations of events s_i in some bounded set A.

When estimated from point process data, the empirical product density function (1.28) provides a description of the density of inter-event distances in an observed point pattern. For instance, high values for small distances are indicative of an overabundance of short inter-event distances (this is a typical situation for cluster processes, where data tend to form groups). Conversely, if short inter-event distances are rare, this will indicate that an inhibitory structure is present, and points tend to separate from each other. In the homogeneous and isotropic case (Cressie (1993)[18]; Stoyan and Stoyan (1994)[23]; Stoyan et al. (1995)[24]), the product density (1.28) depends only on the ***distance*** $r = \| s_i - s_j \|$ between the points s_i and s_j, and thus we write, for the sake of simplicity $l(r)$ for the product density.

If the points in this space have associated the time axis, we have a ***time point process***, and its random points P_i are time instants t_i, which are called the events.

Attention is typically restricted to points in some time interval $[T_0, T_1]$, and to processes with only a finite number of points in any compact subset of $\mathcal{S}$.

Traditionally the points of a point process are thought to be indistinguishable, other than by their times and locations. Often, however, there is other important *"information"* to be stored along with each point. For example, one may wish to analyze a list of points in time and space where a member of a certain species was observed, along with the size or age of the organism, or alternatively a catalog of arrival times and locations of hurricanes along with the amounts of damage attributed to each. Such processes may be viewed as marked spatiotemporal point processes, i.e. random collections of points, where each point has associated with it a further random variable called a mark.

Much of the theory of spatiotemporal point processes carries over from that of spatial point processes. However, the temporal aspect enables a natural ordering of the points that does not generally exist for spatial processes. Indeed, it may often be convenient to view a spatiotemporal point process as a purely temporal point process, with spatial marks associated with each point. Sometimes investigating the purely temporal (or purely spatial) behavior of the resulting marginalized point process is of interest.

The spatial region of interest is often a rectangular portion of $\mathbb{R}^2$ or $\mathbb{R}^3$, but not always. Cases where the points are spatially distributed in a sphere or an ellipse are investigated by Brillinger et al. (1997)[25] and Brillinger (2001)[26]. When the domain of possible spatial coordinates is discrete (e.g. a lattice) rather than continuous, it may be convenient to view the spatiotemporal point process as a sequence $\{N_i\}$ of temporal point processes which may interact with each another.

For modeling and statistical inference purposes we consider point processes in a bounded region of space. Under this restriction it is possible to define point processes by writing down their probability densities.

A point process on the line may be taken as modeling the occurrences of some phenomenon at the time instants t_i with i in some suitable index set. For such a process, there are four equivalent descriptions of the sample paths:

1. counting measures;
2. nondecreasing integer-valued step functions;

3. sequences of points; and
4. sequences of intervals.

In describing a point process as a counting measure, it does not matter that the process is on the real line. However, for the three other methods of describing the process, the order properties of the reals are used in an essential way. While the methods of description may be extended into higher dimensions, they become less natural and, in the case of (4), definitely artificial. We mostly used the intuitive notion of a point process as a counting measure. To make this notion precise, take any subset A of the real line and let $N(A)$ denote the number of occurrences of the process in the set A; i.e.

$$N(A) = \text{ number of indices } i \text{ for which } t_i \text{ lies in } A = \sharp\{i : t_i \in A\}. \tag{1.16}$$

When A is expressed as the union of the disjoint sets $A_1, \ldots, A_r$, say, that is,

$$A = \bigcup_{i=1}^{r} A_i \text{ where } A_i \cap A_j = \emptyset \text{ for } i \notin j,$$

it is a consequence of (1.16) that:

$$N\left(\bigcup_{i=1}^{r} A_i\right) = \sum_{i=1}^{r} N(A_i) \text{ for mutually disjoint } A_1, \ldots, A_r$$

A natural way of measuring the average density of points of a point process is via its mean, or in the case of a stationary point process, its mean density, which Daley and Vere-Jones (2003)[6] define as

$$m = E(N(0, 1]). \tag{1.17}$$

Defining the function

$$M(x) = E(N(0, x]), \tag{1.18}$$

is a consequence of the additivity properties of $N(\cdot)$ as in (1.17), of expectations of sums, and of the stationarity property in (1.18), the following properties for $x, y \geq 0$,

$$\begin{aligned} M(x+y) &= E\left(N(0,x+y]\right) = E\left(N(0,x] + N(x,x+y]\right) \\ &= E\left(N(0,x]\right) + E\left(N(x,x+y]\right) = \\ &= E\left(N(0,x]\right) + E\left(N(0,y]\right) = \\ &= M(x) + M(y) \end{aligned}$$

In other words, $M(\cdot)$ is a nonnegative function satisfying Cauchy's functional equation:

$$M(x+y) = M(x) + M(y) \;\; (0 \leq x,\; y < \infty)$$

Consequently

$$M(x) = M(1)x = m(x) \;\; (0 \leq x < \infty) \tag{1.19}$$

irrespective of whether $M(x)$ is finite or infinite for finite $x > 0$

There is another natural way of measuring the rate of occurrence of points of a stationary point process, due originally to Khintchine (1960)[27].

Proposition 1.17. *For a stationary (or even crudely stationary) point process, the limit:*

$$\lambda = \lim_{h \downarrow 0} \frac{Pr\{N(0,h] > 0\}}{h} \tag{1.20}$$

exists, though it may be infinite.

PROOF:

Introduce the function

$$\phi(x) = Pr\{N(0,x] > 0\}. \tag{1.21}$$

Then $\phi(x) \downarrow 0$ as $x \downarrow 0$, and $\phi(\cdot)$ is subadditive on $(0, \infty)$ because for $x, y > 0$,

$$\begin{aligned}
\phi(x+y) = Pr\{N(0,x+y] > 0\} &= \\
&= Pr\{N(0,x] > 0\} + Pr\{N(0,x] = 0, N(x,x+y] > 0\} \\
&\leq Pr\{N(0,x] > 0\} + Pr\{N(x,x+y] > 0\} = \\
&= \phi(x) + \phi(y) \qquad \square
\end{aligned}$$

Parameter λ is called the intensity of the point process, and when it is finite, (1.20) can be written as:

$$\begin{aligned}
\Pr\{N(x,x+h] > 0\} &= \Pr\{\text{there is at least one point in } (x, x+h]\} \\
&= \lambda h + o(h) \quad (h \downarrow 0)
\end{aligned} \tag{1.22}$$

These two measures of the "rate" of a stationary point process coincide when the point process has the following property (text of proposition, proof and definition from Daley and Vere-Jones (2003)[6]).

Definition 1.18. A point process is **simple** when:

$$\Pr\{N(\{t\}) = 0 \text{ } or \text{ } 1 \text{ } for \text{ } all \text{ } t\} = 1 \tag{1.23}$$

Daley and Vere-Jones (2003)[6] called this sample-path property almost sure orderliness to contrast it with the following analytic property due to Khintchine (1960)[27].

Definition 1.19. A crudely stationary point process is **orderly** when

$$\Pr\{N(0,h] \geq 2\} = o(h) \quad (h \downarrow 0) \tag{1.24}$$

Notice that stationarity plays no role in the definition of a simple point process. In addition, it does not matter whether the point process is defined on the real line or an a Euclidean space (Daley and Vere-Jones (2003)[6]).

Definition 1.20. A regular point process (see Snyder (1975)[28]) is such that the probability of an event occurring in the time interval

$[t, t + \triangle t]$ is given by:

$$\begin{aligned} Pr[\textit{one event in } \ [t, t+\triangle t) | N_t, w_t] &= \mu(t; N_t, w_t)\triangle t \\ Pr[\textit{more than one event in } \ [t, t+\triangle t) | N_t, w_t] &= o(t, \triangle t) \end{aligned} \tag{1.25}$$

where:

N_t is the number of events that have occurred up to time t (observations are assumed to start at time $t = 0$);

w_t is the vector of occurrence times of these N_t events: $w_t = [w_1, \dots, w_{N_t}]$; and

$o(t, \triangle t)$ decreases to zero as $\triangle t$ decreasing faster than linearly: $\lim_{\Delta t \to 0} o(t, \triangle t)/\triangle t = 0$

These equations mean that no more than one event can occur in a sufficiently small interval and that the probability of one event occurring within a small interval, is proportional to the interval's duration. The quantities N_t and w_t describe the ***history*** of the process, giving the number and the times at which all events occurred prior to time t. Note that the probabilities are conditional probabilities: they depend on the point process's history.

Definition 1.21. a) Let $\{N_i\}_{i \in \{1,2,\dots\}}$ be a sequence of nonnegative random variables on some probability space $(\Omega, \mathcal{F}, P)$ such that $0 < N_i \leq N_{i+1}$. Then the sequence $\{N_i\}$ is called a **point process** on $[0, \infty)$. If in addition, $N_i < N_{i+1}$ $\forall i$ then the point process is said to be a **simple point process**.

b) Let $\{N_i\}_{i \in \{1,2,\dots\}}$ be a simple point process on $[0, \infty)$, defined on $(\Omega, \mathcal{F}, P)$, and let $\{Z_i\}_{i \in \{1,2,\dots\}}$ be a sequence of $\{1, 2, \dots, M\}$- valued random variables (also defined on $(\Omega, \mathcal{F}, P)$, with $1 \leq M < \infty$). Then the double sequence $\{N_i, Z_i\}_{i \in \{1,2,\dots\}}$ is called a **M-variate**

point process on $[0, \infty)$. Define for all m, $1 \leq m \leq M$, and all $t \geq 0$

$$V_m(t) = \sum_{i \geq 1} 1(N_i \leq t) 1(Z_i = m). \tag{1.26}$$

Then the M-vector process $V(t) = (V_1(t), V_2(t), \ldots, V_M(t))$ is the M-variate counting process associated with $\{N_i, Z_i\}$.

In our context, N_i will be the occurrence time of the ith market event and Z_i will indicate the event's type. $V_m(t)$ gives the random number of events of type m that have occurred up to and including time t. Because $\{T_i\}$ in the above definition is simple, the possibility of the simultaneous occurrence of two events (of either the same or different types) is ruled out.

Ripley's K-function

The K-function, defined by Ripley (1976)[22], Ripley (1977)[29] is a good indicator for spatial structures (Besag and Diggle (1977)[30], Cressie (1993)[18], Diggle (1983)[31]).

The probability to find a neighbor at a given distance r is very important in applications. The neighbors of point i represents all the points located at a distance less than or equal to a given value r (basically, it represents the number neighbors in a circle of radius r centered on the point i). We denote the expected value by $\nu(r)$. Its estimator, the observed average number of neighbors, is denoted by $V(r)$.

Ripley (1977)[29] showed that:

$$\frac{\nu(r)}{\lambda} = \int_{\rho=0}^{r} g(\rho) 2\pi\rho \, d\rho \tag{1.27}$$

and offers an interpretable measure for the spatial dependence in isotropic stationary point processes.

Thus the K-function is defined as

$$\begin{aligned}\lambda K(r) = & E[\text{ number of extra events within a} \\ & \text{distance } r \text{ of an arbitrary event}]\end{aligned}$$

The K-function provide an interpretable measure of clustering in a point process. The expected number of pairs of events $N(A)$ in a region A with area $\mid A \mid$ with pairwise distance less or equal to r is: $\lambda^2 \mid A \mid K'(r)$. The K-function is a cumulative function and the derivative is another interpretable function called "**the product density function**", defined by:

$$\lambda^{(2)}(r) = \frac{\lambda^2 K'(r)}{2\pi r}, r > 0 \tag{1.28}$$

Definition 1.22. Ripley (1977)[29] defined the K function as:

$$K(r) = \int_{\rho=0}^{r} g(\rho) 2\pi\rho \, d\rho \tag{1.29}$$

where $g(\rho)$ is the pair-correlation function.

If points are distributed independently from each other, $g(\rho) = 1$ for all values of ρ, so $K(r) = \pi r^2$. This value is used as a benchmark:

- $K(r) > \pi r^2$ indicates that the average value of $g(\rho)$ is greater than 1. The probability to find a neighbor at the distance ρ is then greater than the probability to find a point in the same area anywhere in the domain: points are aggregated.
- Inversely, $K(r) < \pi r^2$ indicates that the average neighbor density is smaller than the average point density on the studied domain. Points are then dispersed.

$K(r)$ is estimated by the ratio of the average number of neighbors over the density, the latter being estimated by the total number of points divided by the domain area:

$$\hat{\lambda} = \frac{N}{|A|} \tag{1.30}$$

Thus we have:

$$\hat{K}(r) = \frac{\hat{v}(r)}{\hat{\lambda}} = \frac{V(r)}{N/|A|} \tag{1.31}$$

The average number of neighbors can be expressed more explicitly by defining the indicator $c(i, j, r) = 1$ if the distance between points i and j is at most r, 0 otherwise:

$$\hat{K}(r) = \frac{1}{\hat{\lambda} N} \sum_{i=1}^{N} \sum_{j=1, i \neq j}^{N} c(i, j, r) \tag{1.32}$$

Points located close to the domain borders are problematic because possible neighbors of these points lying outside the domain are not counted. This is so called "edge effect". Ignoring this edge effect results in underestimating K. Ripley (1977)[29] proposed to correct the indicator $c(i, j, r)$ introduced in equation (1.32). We denote L_{ir} the portion of the circle of radius r centered on the point i located inside the domain. If a part of the crown of width dr, inside of which a neighbor is counted, is outside the domain, the neighbor is given a weight equal to the inverse of the ratio between the inside part of the crown ($L_{ir}dr$) and the whole crown ($2\pi r dr$). The idea is that the outside part of the crown could have contained the same neighbor density than the inside part. The correction is then given by:

$$\hat{K}(r) = \frac{1}{\hat{\lambda} N} \sum_{i=1}^{N} \sum_{j=1, i \neq j}^{N} \frac{2\pi r}{L_{jr}} c(i, j, r) \tag{1.33}$$

Cressie (1993)[18] also refers to the K-function as the reduced second-order measure. If we assume the process is completely random then the extra number of events within a distance r will be uniform on a disc. From this we see that:

$$\begin{aligned} \lambda K(r) = \int_0^{2\pi} \int_0^r \{\lambda_2(x)/\lambda\} x \, dx d\theta = \\ = \frac{2\pi}{\lambda} \int_0^r \lambda_2(x) x \, dx \end{aligned} \tag{1.34}$$

and as a result, the second-order intensity function, $\lambda_2(\cdot)$, is:

$$\lambda_2(r) = \frac{\lambda^2}{2\pi r} \frac{\partial K(r)}{\partial r} \tag{1.35}$$

The K-function has many appealing features not shared by $\lambda_2(r)$, such as its invariance to random thinning, physical interpretation, and simple estimation (Cressie (1993)[18]). Several approaches for estimation and interpretation of the K-function are given by Cressie (1993)[18] and Diggle et al. (2003)[32]. However, it must be noted that $K(r)$ does not uniquely determine the distribution of a point process. Different point processes can produce identical K-functions (Baddeley and Silverman (1984)[33]). Furthermore, though the K-function is used to analyze second-order properties of a spatial point pattern, it cannot distinguish between deviations from Complete Spatial Random (CSR) due to lack of uniformity or lack of independence of events. However, since $K(r)$ is defined only for first-order stationary processes, uniformity is a requirement of K-function analysis.

Under CSR, $\lambda_2(r) \equiv \lambda^2$, thus (1.35) reduces to:

$$K(r) = \pi r^2 \tag{1.36}$$

Testing for CSR can be done by comparing the empirical K-function to πr^2. Quite often, inference is based on the L-function defined by:

$$L(r) = \sqrt{K(r)/\pi}, \tag{1.37}$$

and a plot of $L(r) - r$ versus r is used to test for CSR.

Because the probability distribution of the K-function (or the L-function) is intractable, inference about a process is based on simulated K-functions. An envelope is built by simulating the teoretical point process a number of times and defining the K-function for a set of distances r.

For every distance r, $(K_{\mathrm{min}}(r), K_{\mathrm{max}}(r))$ is stored. The upper and lower envelopes are then overlaid on the observed K-function $\hat{K}(r)$.

Inference can be obtained by comparing the observed K- function to the simulated envelope. If $\hat{K}(r) > \bar{K}_{\mathrm{sim}}(r)$ then the number of events within a distance r of an arbitrary event is greater than expected under the hypothesized process. If the hypothesized process is CSR then this would imply an aggregated process if r is small or a regular process if r is large. Conversely, if $\hat{K}(r) < \bar{K}_{\mathrm{sim}}(r)$ then the number of events within a distance r of an arbitrary event is less than expected under the hypothesized process. Reversing our conclusion we would infer that the observed process is regular if r is small or aggregated if r is large. If $\hat{K}(r_0) > \bar{K}_{\mathrm{max}}(r_0)$ or $\hat{K}(r_0) < \bar{K}_{\mathrm{min}}(r_0)$, the hypothesis is rejected for that particular distance.

1.4 Spatial point processes

A spatial point process differs from the first two types of spatial data in that the domain Ω is a random set containing location $\mathbf{s}$ of events. Whereas interest with geostatistical and lattice data lies in studying the properties of $N(\mathbf{s})$ or $E[N(\mathbf{s})]$, for spatial point processes, studying the properties of the set Ω is the primary goal. Note that we can write:

$$N(\mathbf{s}) = \begin{cases} 1, & \text{if } \mathbf{s} \in \Omega \\ 0, & \text{if } \mathbf{s} \notin \Omega \end{cases} \tag{1.38}$$

since locations not in Ω are locations without events.

Definition 1.23. A process where $N(\boldsymbol{s}) = 1, \forall\ \boldsymbol{s} \in \Omega$ is called a **simple point process** to emphasize that only the random locations at which events occur are of interest.

Furthermore, point processes are considered to be orderly in the sense that:

$$\lim P(N(ds) > 1) = 0$$

where ds is an infinitesimal disk at location $\mathbf{s}$ with area (volume) $\mid ds \mid$ and $N(ds)$ denotes the number of events in the disk. In other words, only those processes where any given location can record at most one event are considered.

For geostatistical and lattice data, a weakly stationary process is defined similarly to a second-order (or weakly) stationary temporal process. Specifically, a spatial process is weakly stationary if:

$$E[N(\mathbf{s})] = \mu \tag{1.39}$$

and

$$\mathrm{Cov}[N(\mathbf{s}_i), N(\mathbf{s}_j)] = c(\mathbf{r}) \tag{1.40}$$

where $\mathbf{r} = \mathbf{s}_i - \mathbf{s}_j$ is a two-dimensional vector containing the shift in location from site $\mathbf{s}_i$ to site $\mathbf{s}_j$. This is analogous to weak temporal stationarity in that the stochastic process is location invariant and self-replicating.

For spatial point patterns, weak stationarity is defined through the first and second-order intensities.

Definition 1.24. The **first-order intensity** of a spatial point process is defined as the expected number of points per unit area:

$$\lambda(\boldsymbol{s}) = \lim_{|d\boldsymbol{s}| \to 0} \frac{E[N(d\boldsymbol{s})]}{|d\boldsymbol{s}|} \tag{1.41}$$

Here $d\mathbf{s}$ is an infinitesimal region containing the site $\boldsymbol{s}$, $N(d\mathbf{s})$ represents the number of points located in $d\mathbf{s}$, and $|d\mathbf{s}|$ is the area (volume) of the region $d\mathbf{s}$. Throughout the text, the notation $|\cdot|$ will represent the Lesbesgue measure on the spatial region Ω.

Definition 1.25. The **second-order intensity** is a measure of the dependency structure of the events in Ω and is given by:

$$\lambda_2(\mathbf{s}_i, \mathbf{s}_j) = \lim_{\substack{|d\boldsymbol{s}_i| \to 0 \\ |d\boldsymbol{s}_j| \to 0}} \frac{E[N(d\boldsymbol{s}_i)N(d\boldsymbol{s}_j)]}{|d\boldsymbol{s}_i||d\boldsymbol{s}_j|} \tag{1.42}$$

The second-order intensity $\lambda_2(s_i, s_j)$ contains information about the stochastic dependence between events in two regions. Although $\lambda_2(\cdot,\cdot)$ is akin to $c(\cdot)$ defined previously, it is not a covariance function.

As with the other types of spatial data, a spatial point process is weakly stationary if the process is location invariant. This is equivalent to saying $\lambda(\mathbf{s}) \equiv \lambda$ so that the expected number of events at an arbitrary location $\mathbf{s}$ is constant for all $\mathbf{s} \in D$; and $\lambda_2(\mathbf{s}_i, \mathbf{s}_j) = \lambda_2(\mathbf{r})$ so that the dependence between events at two arbitrary locations $\mathbf{s}_i$ and $\mathbf{s}_j$ depends only on the distance $\mathbf{r}$.

A spatial process has the distinction from a temporal process in that its observations typically cannot be ordered. However, if a spatial process is weakly stationary, we can write the covariance between any two observations as a function of the distance between them. We can define two types of weakly stationary covariance functions: anisotropic and isotropic ones. An anisotropic process has covariance function defined by (1.40) but covariances that differ with direction. In an isotropic process the covariance function does not depend on direction and $\mathbf{r}$ can be replaced by:

$$r = \| \mathbf{s}_i - \mathbf{s}_j \| = [(\mathbf{s}_{ix} - \mathbf{s}_{jx})^2 + (\mathbf{s}_{iy} - \mathbf{s}_{jy})^2]^{1/2} \tag{1.43}$$

the Euclidean distance between $\mathbf{s}_i$ and $\mathbf{s}_j$. An isotropic process is thus invariant under coordinate shifts and rotations.

Definition 1.26. ***Stationarity and isotropy*** mean that:

$$N + s = \{x_i + s\} \text{ and } rN = \{rx_i\}$$

have the same distribution as N for any $s \in R^d$ and any (Euclidean) rotation r around the origin, respectively.

Spatial point patterns can be classified into three classes:
- Regular processes
- Aggregated processes
- Complete spatial randomness

All three types can be observed with isotropic or anisotropic covariance functions (Cressie (1993)[18]).

1.4.1 Regular processes

The simplest type of regular process is one that does not allow two events to be within a distance r of each other (Cressie (1993)[18]). Other subclasses of regular processes are listed by Cressie (1993)[18] and back in time referred in Matern (1986)[34], Stoyan and Stoyan (1994)[23], and Bartlett (1975)[19].

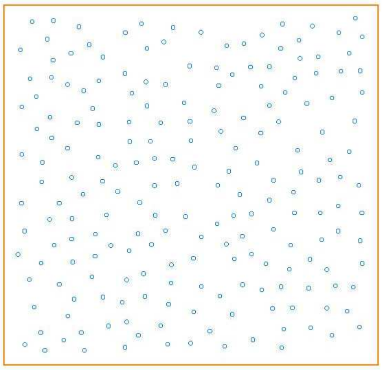


FIGURE 1.2: A sequential inhibition processes (Diggle, 1976)

An example of a regular pattern in shown in Figure 1.2. Two hundred events were generated using an inhibition radius ($r = 0.05$) on a unit square.

1.4.2 Cluster processes

Aggregated, or clustered, processes include the Poisson cluster process, the Neyman- Scott process, and the Cox process. A general aggregated point pattern can be thought of a parent-offspring process where offsprings are dispersed around a parent event. The Poisson cluster process is generated by first obtaining parent events from a homogeneous Poisson process with mean measure μ_p, where μ_p is the expected number of parents. Each parent event produces a random number of offspring positioned around the parent according to some bivariate probability density. The parent events are then removed leaving only the offspring point process.

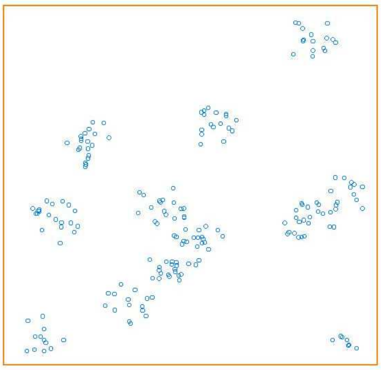


Figure 1.3: A Poisson cluster process

Figure 1.3 shows an example of two-hundred events in a Poisson cluster process on a unit square with $\mu_p = 15$ and a bivariate normal distribution with radius 0.075 for the offsprings events according to a standard bivariate normal distribution.

1.4.3 Complete spatial randomness

A CSR process is one that generates events uniformly and independently in a region A with area $|A|$. The number of events $N(A)$, $A \subset \Omega$, follows a Poisson distribution with mean $\lambda|A|$, where λ is the average number of events per unit area. Further, given n events $\mathbf{s}_i \in A$, the $\mathbf{s}_i$ are independent realizations from a uniform distribution on A.

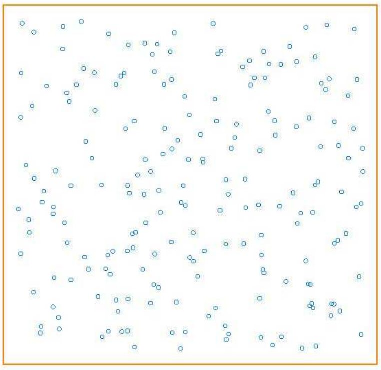


FIGURE 1.4: A CSR pattern

Figure 1.4 show a CSR process with $n = 200$. Note that conditioning on n, yields a Binomial process. The importance of the CSR hypothesis lies in the fact that it is often used as a null hypothesis for testing a spatial point pattern. Under CSR, a spatial point pattern has no structure and thus failing to reject such a hypothesis warrants no further examination of the data (Diggle (1983)[31]). A CSR spatial point pattern implies uniformity of events ($E[N(A)] = \lambda|A|$) as well as independence:

$$\text{Cov}[N(A), N(B)] = 0 \text{ if } A \cap B = \emptyset$$

1.5 Marked spatial point processes

A flexible marked point process of the form:

$$N_m = \{[x_i; m(x_i)]\}$$

can be built starting from an unmarked point process $N = \{x_i\}$ and providing each point $x_i \in N$ by a real-valued mark $m(x_i)$. This procedure is called ***"marking"***.

Definition 1.27. Given a point process N a marked point process $N_m = \{[x_i; m(x_i)]\}$ is obtained if each point $x_i \in N$ is provided with a random variable $m(x_i)$ called ***mark***.

The focus here is in the intensity-dependence which means that the local point density affects the marks. For example, intensity-dependence allows the marks to be large (small) in areas of low point intensity and small (large) in areas of high intensity.

The simplest marking strategy is *independent marking*, where the marks are drawn for each point x_i from a probability distribution independent of each other and independent of the point process. This model is often used as a reference model. In geostatistical marking (Mase (1996)[35]; Schlather et al. (2004)[36]; Illian et al. (2008)[37]) the marks are drawn from a random field $\{U(s)\}$ which is independent of the point process N: the marks are $m(x_i) = U(x_i)$. This marking generates correlated marks but is not able to model intensity-dependence by construction.

A step forward is intensity-dependent marking suggested by Stoyan (2008)[38] and Myllymäki (2006)[39] for the stationary log Gaussian Cox process generated by a random intensity $\{\Lambda(s)\}$. In these markings the mean of the conditional mark distribution of $m(x_i)$ given $\Lambda(x_i)$ is a function of $\Lambda(x_i)$ and the marks are conditionally independent given the intensity $\{\Lambda(s)\}$. Although the marks are conditionally independent, they are marginally correlated. The log Gaussian Cox process as a point process model is a natural choice for two reasons. First, intensity-dependent marking presumes the existence of local variation in the point intensity, and thus, only clustered or heterogeneous point process models are relevant. Cox processes are such models, see Møller et al. (1998)[40] and Møller and Waagepetersen (2004)[41]. Second, the log Gaussian Cox process is a flexible model with nice theoretical properties. The existing intensity-dependent markings are useful models but assume that the variance of the conditional mark distribution does not depend on the point intensity.

In this case the intensity function $\lambda = E(\Lambda(s))$, that gives the mean number of points per unit volume, can be used to write

$$\lambda^2 g(\| o - r \|) \, do dr$$

which gives the probability that two infinitesimal disjoint regions of volumes do and dr both contain exactly one point of N. For further details see e.g. Stoyan and Stoyan (1994)[23], Stoyan et al. (1995)[24] or Illian et al. (2008)[37].

Stationarity means that the translated process $\{[x_i + s; m(x_i)]\}$ has the same distribution as N_m. Note that the marks are kept untouched in the translation. Isotropy is defined in a similar way when translation is replaced by rotation around the origin.

Various first-order and second-order mark characteristics have been suggested to describe the properties of marked point processes (Stoyan and Stoyan (1994)[23]; Stoyan et al. (1995)[24]; Schlather (1999)[42]; Schlather et al. (2004)[36] and Illian et al. (2008)[37]). Their empirical counterparts are used in model identification, parameter estimation, evaluation of goodness-of-fit and in model interpretation.

These mark characteristics are conditional quantities (in the Palm sense): let E_x and var_x stand for the conditional expectation and variance, respectively, given that there is a point of N at the location x. Further let E_{xy} refer to the conditional expectation given there are two points of N at locations x and y. Because of stationarity and isotropy, it suffices to consider expectations E_o and E_{or} with $\| r \|= r$.

Then, the mean mark $\mu_m = E_o(m(o))$ and the mark variance $\sigma_m^2 = \mathrm{Var}_o(m(o))$, which are the mean and variance of the mark distribution function $\mathcal{F}_M(m)$, are first-order characteristics of marks and conditional on *"there is a point of N at o"*.

1.6 Poisson processes

Many processes in everyday life that "count" events up to a particular point in time can be accurately described by the so-called Poisson process, named after the French scientist Siméon Poisson (1781-1840; appointed as full professor at the Ecole Polytechnique, Paris, in 1806 as a successor of Fourier). An (ordinary) Poisson process is a special Markov process, in continuous time, in which the only possible jumps are to the next higher state.

The simplest Poisson process is the stationary Poisson process on the line who is completely defined by the following equation, in which we use $N(a_i, b_i]$ to denote the number of events of the process falling in the half-open interval $(a_i, b_i]$ with $a_i < b_i \leq a_{i+1}$:

$$Pr\{N(a_i, b_i] = n_i, i = 1, \ldots, k\} = \prod_{i=1}^{k} \frac{[\lambda(b_i - a_i)]^{n_i}}{n_i!} e^{\lambda(b_i - a_i)}. \quad (1.44)$$

This definition embodies three important features:
(i) *the number of points in each finite interval* $(a_i, b_i]$ *has a Poisson*

distribution;
(ii) *the number of points in disjoint intervals are independent random variables; and*
(iii) *the distributions are stationary: they depend only on the lengths* $b_i - a_i$ *of the intervals.*

Thus, the joint distributions are multivariate Poisson of the special type in which the variates are independent.

Let us first summarize a number of properties that follow directly from above. The ***mean*** $M(a,b]$ and ***variance*** $V(a,b]$, of the number of points falling in the interval $(a,b]$ are given by:

$$M(a,b] = \lambda(b-a) = V(a,b]. \tag{1.45}$$

The constant λ here can be interpreted as the *mean rate* or *mean density* of points of the process. It also coincides with the *intensity* of the point process.

The fact that the mean and variance are equal and that both are proportional to the length of the interval provide a useful diagnostic test for the stationary Poisson process: estimate the mean $M(a,b]$ and the variance $V(a,b]$ for half-open intervals $(a,b]$ over a range of different lengths, and plot the ratios $V(a,b]/(b-a)$. The estimates should be approximately constant for a stationary Poisson process and equal to the mean rate. Any systematic departure from this constant value indicates some departure either from the Poisson assumption or from stationarity (see Cox and Lewis (1966)[43] (Section 6.3) for more discussion).

A Poisson process may also be viewed as a counting process that has particular, desirable, properties. A counting process $\{N(t), t \geq 0\}$ is a stochastic process that counts the number of events that have occurred up to time t. Obviously, $N(t)$ is non-negative and integer-valued for all $t \geq 0$. Furthermore, $N(t)$ is nondecreasing in

t. $N(t_i) - N(t_{i+1})$ equals the number of events in the time interval $(t_i, t_{i+1}], t_i < t_{i+1}$.

$N(t)$ could denote the number of arrivals of customers at a railway station in $(0, t]$, or the number of accidents on a particular highway in that time interval, or the number of births of animals in a particular zoo in $(0, t]$, or the number of calls to a telephone call-center during that period. A Poisson process is a counting process that has the desirable additional properties that the number of events in disjoint intervals are independent ("independent increments") and that the number of events in any given interval depends only on the length of that interval, and not on its particular position in time ("stationary increments"). In the case of the arrivals at the railway station, the stationarity assumption is clearly not fulfilled; there will be many more arrivals between 5 p.m. and 6 p.m. than between, say, 5 a.m. and 6 a.m. Still, one might wish to study the arrival process at the railway station during the rush hour. Restricting oneself to subsequent working days between 5 p.m. and 6 p.m. does allow one to use the stationary increments assumption.

Definition 1.28. A Poisson process $N(t), t \geq 0$ is a counting process with the following additional properties:
(i) $\mathrm{N}(0) = 0$.
(ii) The process has stationary and independent increments.
(iii) $P(N(h) = 1) = \lambda h + o(h)$ and $P(N(h) \geq 2) = o(h), h \downarrow 0$, for some $\lambda > 0$.
Above, the $o(h)$ symbol indicates that the ratio

$$\frac{P(N(h) \geq 2)}{h} \text{ tends to zero for } h \downarrow 0$$

The last property may look awkward at first sight, but is insightful. It states that having two or more events in a small time interval is extremely unlikely, while the probability of a single event is approximately proportional to the length of that small interval. An equivalent definition is the following.

Definition 1.29. A Poisson process $\{N(t), t \geq 0\}$ is a counting process with the following additional properties:
(i) N(0) = 0.
(ii) The process has stationary and independent increments.
(iii) $P(N(t) = n) = e^{-\lambda t}\frac{(\lambda t)^n}{n!}$, $n = 0, 1, 2, \ldots$.

Of course, the last property states that the number of events in any interval of length t is Poisson distributed with mean λt. λ is called the rate of the Poisson process.

It readily follows that the Probability Generating Function of $N(t)$ is given by:

$$E[z^{N(t)}] = \Sigma_{n=0}^{\infty}\, z^n P(N(t) = n) = e^{-\lambda(1-z)t}.$$

Differentiation yields:

$$E[N(t)] = \lambda t$$
$$E[N(t)(N(t) - 1)] = (\lambda t)^2$$

and hence:

$$\text{Var}(N(t)) = \lambda t.$$

We close this section by giving yet another two equivalent definitions of the Poisson process, and we enumerate three types of Poisson Processes.

Definition 1.30. A Poisson process $\{N(t), t \geq 0\}$ is a counting process with the following additional properties:
(i) N(0) = 0
(ii) The only changes in the process are unit jumps upward. The

intervals between jumps are independent, exponentially distributed random variables with mean $1/\lambda, \lambda > 0$.

Definition 1.31. A Poisson process is a continuous-time counting processes $\{N(t), t \geq 0\}$ that possesses the following properties:
(i) N(0) = 0.
(ii) **Independent increments:** in disjoint intervals the numbers of occurrences counted are independent from each other.
(iii) **Stationary increments:** the probability distribution of the number of occurrences counted in any time interval only depends on the length of the interval.
(iv) No counted occurrences are simultaneous.

1.6.1 Homogeneous Poisson process

A homogeneous Poisson process is characterized by a rate parameter λ, also known as intensity, such that the number of events in time interval $(t, t+\tau]$ follows a Poisson distribution with associated parameter $\lambda\tau$. This relation is given as:

$$P[(N(t+\tau) - N(t)) = k] = \frac{e^{-\lambda\tau}(\lambda\tau)^k}{k!} \; k = 0, 1, \ldots \tag{1.46}$$

where $N(t+\tau) - N(t)$ describes the number of events in time interval $(t, t+\tau]$. Just as a Poisson random variable is characterized by its scalar parameter λ, a homogeneous Poisson process is characterized by its rate parameter λ, which is the expected number of “events” or “arrivals” that occur per unit time.

1.6.2 Non-homogeneous (inhomogeneous) Poisson process

In general, the rate parameter may change over time. In this case, the generalized rate function is given as $\lambda(t)$. Now the expected number of events between time a and time b is:

$$\lambda_{a,b} = \int_a^b \lambda(t)dt \tag{1.47}$$

Thus, the number of arrivals in the time interval $(a, b]$, given as $N(b) - N(a)$, follows a Poisson distribution with associated parameter $\lambda_{a,b}$

$$P[(N(b) - N(a)) = k] = \frac{e^{-\lambda_{a,b}}(\lambda_{a,b})^k}{k!} \; k = 0, 1, \ldots \tag{1.48}$$

A homogeneous Poisson process may be viewed as a special case when $\lambda(t) = \lambda$, a constant rate.

1.6.3 Spatial Poisson processes

A further variation of the Poisson process, called the spatial Poisson process, introduces a spatial dependence on the rate function and is given as $\lambda(\overrightarrow{x}, t)$ where $\overrightarrow{x} \in V$ for some vector space V. For any set $S \in V$ (a spatial region) with finite measure, the number of events occurring inside this region can be modeled as a Poisson process with associated rate function $\lambda_S(t)$ such that:

$$\lambda_S(t) = \int_S \lambda(\overrightarrow{x}, t)d\overrightarrow{x}$$

In the special case that this generalized rate function is a separable function of time and space, we have:

$$\lambda(\overrightarrow{x}, t) = f(\overrightarrow{x})\lambda(t)$$

for some function $f(\overrightarrow{x})$. Without loss of generality, we assume that this property is satisfied:

$$\int_V f(\overrightarrow{x})d(\overrightarrow{x}) = 1$$

(If this is not the case, $\lambda(t)$ can be scaled appropriately.)

$f(\overrightarrow{x})$ represents the spatial probability density function of these random events in the following sense. Sampling this spatial Poisson process is equivalent to sampling a Poisson process with rate function $\lambda(t)$, and associated with each event a random vector $\overrightarrow{x}$ sampled from the probability density function $f(\overrightarrow{x})$.

1.7 Cluster processes

Clustering often occur in application such as botany, where a plant sets seed around itself, resulting in a cluster of plants of this species the following year.

This clustering mechanism is the idea behind the construction of Neyman-Scott points processes (Stoyan et al. (1995)[24]; Diggle and Chetwynd (1991)[44]) where each parent point in a stationary Poisson Process of parent point gives rise to a stochastic number of offspring points, independently distributed around the parent points according to a specified density f. The cluster mechanism is also a natural way to describe the locations of individuals from consecutive generations of a branching process, an application with unexpectedly rich mathematical structure as well as its obvious practical applications.

The intuitive motivation of such processes involves two components: the locations of clusters and the locations of elements within a cluster. The superposition of the latter constitutes the "observed" process. To model the cluster elements, we specify a countable family of point processes $N(\cdot|y_i)$ indexed by the cluster centers $\{y_i\}$. To model the cluster locations, we suppose there is given a process N_c of cluster centers, is given often unobserved, whose generic realization consists of the points $\{y_i\} \subset Y$. The centers y_i act as the germs (ancestors in the branching process context) for the clusters they generate; it is supposed in general that there are no special features attached to the points of a given cluster that would allow them to be distinguished from the points in some other cluster. More formally, we have the following definition.

Definition 1.32. N is a cluster process on the complete separable metric space (CSMS) X, with center process N_c on the CSMS Y and a component processes the measurable family of point processes $\{N(\cdot|y) : y \in Y\}$, when for every bounded $A \in \mathcal{B}_X$,

$$N(A) = \int_Y N(A|y)N_c(dy) = \sum_{y_i \in N_c(\cdot)} N(A|y_i) < \infty \tag{1.49}$$

The definition requires the superposition of the clusters to be almost surely finite. There is, however, no requirement in general that the individual clusters must themselves be a.s. finite (i.e. the condition $N(X|y) < \infty$ a.s. is not necessary), although it is a natural constraint in many examples. A general cluster random measure can be introduced in the same way by allowing the component processes to be random measures.

For the remainder of this section, we require the component processes to be mutually independent. We shall then speak of the component processes as coming from an *independent measurable family* and thereby defining an *independent cluster process.* In this definition, it is to be understood that multiple independent copies of $N(\cdot|y)$ are taken when $N_c\{y\} > 1$. If $Y = X$ (i.e. the cluster centre process and the component processes are all defined on the same space X, and X admits translations), then the further constraint that the translated components $N(A-y|y)$ are identically distributed may be added, thus producing a natural candidate for a stationary version of the process.

Conditions for the existence of the resultant point process are not so easily obtained as for the Cox process, even though the superposition of the cluster member processes involves only operations that are clearly measurable. The difficulty revolves around the finiteness requirement embodied in equation (1.49). The number of clusters that are potentially able to contribute points to a given bounded set soar as the dimension of the state space increases, imposing delicate constraints that have to be met by any proposed existence theorem. For independent cluster processes, the finiteness condition can be rephrased somewhat more formally.

Lemma 1.33. *An independent cluster process exists if and only if, for any bounded set* $A \in \mathcal{B}_X$ *,*

$$\int_Y p_A(y) N_c(dy) = \sum_{y_i \in N_c} p_A(y_i) < \infty, \quad \prod_c\text{-a.s.}, \tag{1.50}$$

where $p_A(y) = \Pr\{N(A|y) > 0\}$ *for* $y \in Y$ *and* $A \in \mathcal{B}_X$ *, and* $\prod_c$ *is the probability measure for the process of cluster centres.*

PROOF:

The sum (1.50) is required to converge a.s. as part of the definition of a cluster process. The converse, for given N_c, is an application of the second Borel Cantelli lemma to the sequence of events:

$$E_i = \{\text{cluster } i \text{ contributes at least one point to the set } A\}$$

$\square$

The condition of Lemma 1.33 can alternatively be rephrased in terms of generating functionals. When the components of the process are stationary (i.e. their cluster centre process is stationary and the distribution of the cluster members depends only on their positions relative to the cluster centre), a simple sufficient condition for the resultant cluster process to exist is that the mean cluster size be finite.

The moments are easier to handle. Thus, taking expectations conditional on the cluster centres yields:

$$E[N(A)|N_c] = \sum_{y_i \in N_c} M_1(A|y_i) = \int_Y M_1(A|y)N_c(dy), \tag{1.51}$$

where $M_1(E|y)$ denotes the expectation measure of the cluster member process with centre at y, assuming this latter exists. From the assumption that the cluster member processes form a measurable family, it follows also that whenever $M_1(A|y)$ exists, it defines a *measurable kernel* (a measure in A for each y and a *measurable function* of y for each fixed Borel set $A \in \mathcal{B}_X$). Then we can take expectations with respect to the cluster centre process to obtain:

$$E[N(A)] = \int_Y M_1(A|y)M^c(dy), \tag{1.52}$$

finite or infinite, where $M^c(\cdot) = E[N_c(\cdot)]$ is the expectation measure for the process of cluster centers. From this representation, it is clear that the first-moment measure of the resultant process exists if and only if the integral in (1.52) is finite for all bounded Borel sets A.

Similar representations hold for the higher-order moment measures. In the case of the second factorial moment measure, for example, we need to consider all possible ways in which two distinct points from the superposition of clusters could fall into the product set $A \times B(A, B \in \mathcal{B}_X)$. Here there are two possibilities: either both points come from the same cluster or they come from distinct clusters. Incorporating both cases, supposing the cluster centre process is given, we obtain:

$$
\begin{aligned}
E[N^{[2]}(A \times B|N_c)] = & \int_Y M_{[2]}\,(A \times B|y)\, N_c(dy) + \\
& + \int_{Y^{(2)}} M_1(A|y_1) M_1(B|y_2) N_c^{[2]}(dy_1 \times dy_2),
\end{aligned}
$$

where the superscript in $N^{[2]}$ denotes the process of distinct pairs from N and in the second integral we have used the assumption of independent clusters. Taking expectations with respect to the cluster centre process, we obtain for the second factorial moment of the cluster process:

$$
\begin{aligned}
E\big[N^{[2]}(A \times B)\big] = & \int_Y M_{[2]}\,(A \times B|y)\, M^c(dy) + \\
& + \int_{Y^{(2)}} M_1(A|y_1) M_1(A|y_2) M_{[2]}^c(dy_1 \times dy_2),
\end{aligned}
\tag{1.53}
$$

Again, the second factorial moment measure of the cluster process exists if and only if the component measures exist and the integrals in (1.53) converge. Restated in terms of the factorial cumulative measure, equation (1.53) reads:

$$
\begin{aligned}
C_{[2]}(A \times B) = & \int_Y^{(2)} M\,(A|y_1)\, M(B|y_2) C_{[2]}^c(dy_1 \times dy_2) + \\
& + \int_Y M_{[2]}(A \times B|y) M^c(dy).
\end{aligned}
\tag{1.54}
$$

Many of these relationships are derived most easily, if somewhat mechanically, from the portmanteau relation for the probability generating functionals, which takes the form, for $h \in \mathcal{V}(X)$ and exploiting the independent cluster assumptions,

$$
\begin{aligned}
G[h] = E(G[h|N_c]) = E\left[\exp\left(-\int_Y (-\log G_m[h|y]) N_c(dy)\right)\right] = \\
= G_c\left[G_m\left[h|\cdot\right]\right],
\end{aligned}
$$

where $G_m[h|y]$ for $h \in \mathcal{V}(X)$ is the p.g.fl. of $N(E|y)$, and

$$G[h|N_c] = \prod_{y_i \in N_c} G_m[h|y_i] = \exp\left[-\int_Y (-\log G_m[h|y])N_c(dy)\right]$$

is the conditional p.g.fl. of N given N_c. The a.s. convergence of the infinite product in the last equation is equivalent to the a.s. convergence of the sum in Lemma 1.33. The measurable family requirements of the family of p.g.fl.s for the cluster centers follow from the initial assumptions for the process. Thus, the p.g.fl. representation is valid whenever the cluster process exists.

One class of cluster processes occurs so frequently in applications, and is so important in the theory, that it warrants special attention. If in this class, the cluster centers are the points of a Poisson process, we speak of a Poisson cluster process, and the clusters are independent and finite with probability 1.

1.8 Spatio-temporal point processes

Spatiotemporal point processes will be considered as a hybrid of spatial and temporal components. Extending the definition of spatial processes $N(\mathbf{s})$ defined in formula (1.20) to include time, we obtain the following definition of a ***spatio-temporal point process:***

$$\{N(\mathbf{s}, t) : \mathbf{s} \subseteq \Omega(t) \subset \mathbb{R}^2, t \in T\} \tag{1.55}$$

where $\Omega(t)$ is the spatial index at time t. Notice that a spatiotemporal process is a composite of a spatial and temporal process. This distinction is critical, since time flows unidirectionally, whereas space is not directed.

Spatiotemporal point processes have been studied thoroughly in the context of earthquake data. Ogata (1999)[45] wrote a summary paper of parametric, maximum likelihood techniques. Choi and Hall (1999)[46], Choi and Hall (2000)[47] added nonparametric estimators of the intensity function using a kernel estimator approach and discuss asymptotic theory for many of the parametric estimators. Rathbun (1993)[48] and Rathbun and Cressie (1994)[49] discuss spatiotemporal point processes in the context of tree growth. In their paper, they present methods of parametric estimation by splitting the process into three components: a spatial birth process, a spatial growth process, and a space-time survival process.

There also exists literature on spatiotemporal processes for lattice and geostatistical data. See, for example, Kooperberg and OSullivan (1996)[50] and Haas (1995)[51]. Haas (1995)[51] develops the idea of a spatiotemporal covariance function through the construction of spatiotemporal cylinders for use in prediction of wet sulfate deposition (a geostatistical process). Essentially, Haas creates a cylinder centered at location $\mathbf{s}_0 = (x_0, y_0)$ with length $m_T = t_U - t_L$ for a user-defined time interval (t_L, t_U). He then uses n_c, the number of points located inside the cylinder, to make a prediction of wet sulfate deposition at the location $\mathbf{s}_0$.

Here, we have separated the spatiotemporal point processes into four distinct scenarios, all this inspired from nature:

- the earthquake process,
- the explosion process,
- the birth-death process,
- and point patterns sampled in time.

1.8.1 Earthquake processes

The type of data that would be observed from an earthquake process corresponds to the location $\mathbf{s}$ where an event occurred at time t. Characteristic for this process is that it is orderly in both space and time: only one event can occur at a particular location and time. Analysis of earthquake processes are discussed extensively in the geophysical literature: Rathbun (1996)[52], Ogata (1999)[45], Choi and Hall (1999)[46]. Depending on whether we record ***"when" (time) and "where" (space)*** an earthquake occurred or also its magnitude, the process is **simple** or **marked**. Cressie (1993)[18] refers to this type of space-time process as a space-time shock process because events occur simultaneously over time and space. The realization of such a process consists of locations $\mathbf{s}_i \in \Omega^*$ at times $t_i \in T^*, i = 1, 2, ...,$ where Ω^* and T^* are realizations of a particular point pattern (see Figure1.5).

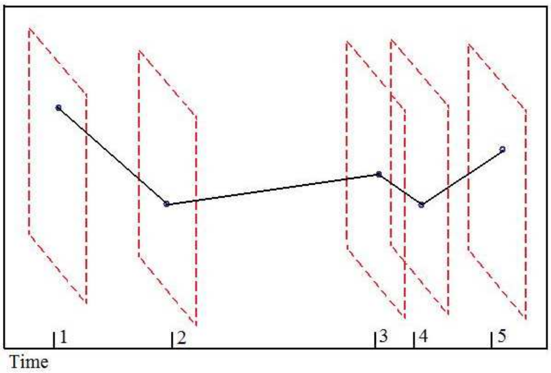


FIGURE 1.5: An earthquake process

Integration of events over a time interval $[0,\ t]$ results in a spatial point pattern that can be analyzed by standard methods for such data. Cressie (1993)[18] suggests that this type of space-time process can be viewed as a ***marked spatial point process*** with mark space T^*.

1.8.2 Explosion processes

The basic explosion process combines a point process in space with a point process in time. Realizations of such a process consist of locations $\mathbf{s}_{i1}, ..., \mathbf{s}_{in_i} \in \Omega^*$ at times $t_i \in T^*$ (see Figure1.6).

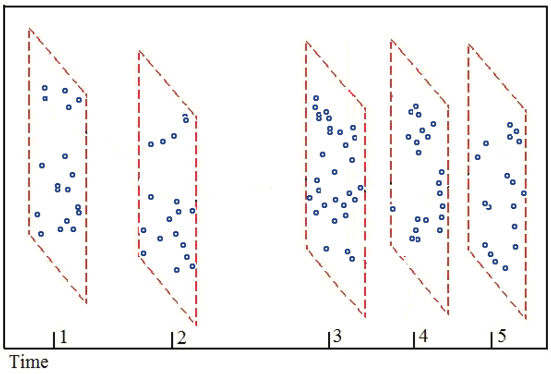


FIGURE 1.6: An explosion process

The name relates to the type of data that might be observed from this process. At each time t a scatter of points with intensity $\lambda(\mathbf{s})$ is observed. Furthermore, the explosions occur as a point process with intensity $\lambda(t)$. In other words, a temporal event triggers the generation of a spatial point pattern.

One example of such a process is when temporal events occur completely at random and $N(\mathbf{s}_i, t_i \in T^*)$ is a sequential inhibition process.

1.8.3 Birth-death processes

This process is typically used to model objects that are born at some random location and live for a random time. Cressie (1993)[18] refers to these as space-time survival point processes. See, for example, Rathbun and Cressie (1994)[49]. Realizations of birth-death processes often consist of locations of events at fixed points in time $t_1, ..., t_n$. Key to these processes is that events remain at the same location during their lifetime, in contrast to, e.g., a Brownian motion where *particles move through a volume* (see Figure1.7).

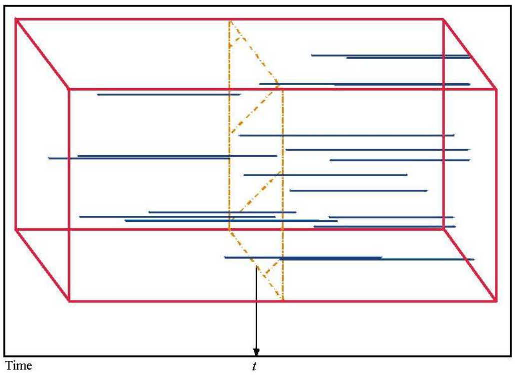


FIGURE 1.7: A birth-death process

1.8.4 Point patterns sampled in time

This process consists of T spatial point patterns observed at equally or unequally spaced time intervals. In contrast to the explosion process where $t_i \in T^*$ represent a complete *"mapping"* of the time points at which temporal events occurred, the sample times in a sequence of sampled point patterns are fixed by a probability or non-stochastic sample design. At each time point the spatial locations of events are *"completely recorded"* in the domain of interest, i.e., the **spatial point pattern is mapped** (see Figure1.8).

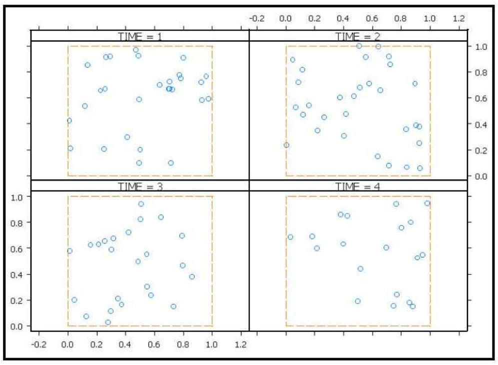


FIGURE 1.8: Points patterns sample in time

A sampled point pattern can be indistinguishable from a birth-death process with fixed points in time. Whereas birth-death processes are excellent models to describe the survival of stationary objects, a sampled point pattern can also be thought of as recording the locations of non-stationary objects that move through space and time.

If an event is observed at locations $\mathbf{s}^*$ at time t_{i-1} but not at t_i could be due to the death of a stationary object that lived at $\mathbf{s}^*$ at t_{i-1} or the displacement of a non-stationary object that occupies a different location at t_i. Combinations of these processes are obvious. In the study of animals, for example, individuals are born between t_{i-1} and t_i, their locations are observed at times $t_i, t_{i+1}, ..., t_{j-1}$, die between times t_{j-1} and t_j and are no longer recorded at sample times $t \geq t_j$. The inference of interest in sampled point patterns depends on the nature of the process as pure birth-death, continuous motion, or a mixture thereof. Regardless of the genesis, however, studying the inter-relationship among point patterns over time is common to all sampled point patterns.

1.8.5 Intensity measures of spatio-temporal point patterns (First-Order Properties)

Any analytic spatial temporal point process is uniquely characterized by its associated conditional rate process λ (Fishman and Snyder (1976)[53]). $\lambda(\mathbf{s}, t)$ may be thought of as the frequency with which events are expected to occur around a particular location $(\mathbf{s}, t)$ in space-time, conditional on the prior history H_t of the point process up to time t. Note that in the statistical literature (e.g. Daley and Vere-Jones (2003)[6]; Karr (1991)[54]), λ is more commonly referred to as the conditional intensity rather than conditional rate. The idea is to express the relationship of points not only by distance but by time lag. Our initial concern is to modify the first- and second-order intensity functions to incorporate temporal and spatial dimensions.

The intensity function $\lambda(\cdot)$ can be modified to include a time component as:

$$\lambda(\mathbf{s}, t) = \lim_{\substack{|d\mathbf{s}| \to 0 \\ |dt| \to 0}} \frac{E[N(d\mathbf{s}, dt)]}{|d\mathbf{s}||dt|} \tag{1.56}$$

where $N(A)$ represents the number of events in volume A, $d\mathbf{s}$ is an infinitesimal disk containing the location $\mathbf{s}$, and dt is an infinitesimal interval containing time t. Following Haas (1995)[51] we consider A as a cylinder with face $d\mathbf{s}$ and height dt. Note that this is an obvious extension of the first-order intensity in a spatial point process which considers the expected number of points in a disc ds. Thus, $N(ds, dt)$ represents the number of points in the cylinder. The limit in (1.56) shrinks the cylinder around the point $(\mathbf{s}, t)$.

Definition 1.34. The **marginal spatial intensity** is defined as:

$$\lambda(\mathbf{s}, \cdot) = \int_T \lambda(\mathbf{s}, t) dt \tag{1.57}$$

where the integration is over all times in T.

Definition 1.35. The **marginal temporal intensity** is defined as:

$$\lambda(\cdot, t) = \int_A \lambda(\mathbf{s}, t) ds \tag{1.58}$$

where $\int_A$ represents integration over the region A.

The marginal intensities allow us to view one component, either spatial or temporal, while ignoring the other component. Furthermore, if $\lambda(\mathbf{s}, t)$ is equal to either or both of the marginal intensities then the process is said to be first-order stationary in time, space, or both.

Conditioning on either location or time allows us to isolate either component in a spatiotemporal process. Conditioning on t reduces the spatiotemporal intensity defined in (1.56) to ***conditional spatial intensity:***

$$\lambda(\mathbf{s}|t=t^*) = \lim_{|d\mathbf{s}|\to 0} \frac{E[N(d\mathbf{s},t^*)]}{|d\mathbf{s}|} \tag{1.59}$$

Similarly, conditioning on $\mathbf{s}$ leads to the ***conditional temporal intensity:***

$$\lambda(t|\mathbf{s}=\mathbf{s}^*) = \lim_{|dt|\to 0} \frac{E[N(\mathbf{s}^*,dt)]}{|dt|} \tag{1.60}$$

Definition 1.36. The **average spatial intensity** represent the average intensity over the interval dt at location $\boldsymbol{s}$:

$$\lambda(\mathbf{s},dt) = \lim_{|d\mathbf{s}|\to 0} \frac{E[N(d\mathbf{s},dt)]}{|d\mathbf{s}||dt|} \tag{1.61}$$

Definition 1.37. The **average temporal intensity** represent the average intensity over the space $|d\boldsymbol{s}|$ at time t:

$$\lambda(d\mathbf{s},t) = \lim_{|dt|\to 0} \frac{E[N(d\mathbf{s},dt)]}{|d\mathbf{s}||dt|} \tag{1.62}$$

This notation will allow us to collapse (1.56) one component at a time: spatially or temporally. Also note that:

$$\lambda(\mathbf{s},t) = \lim_{|dt|\to 0} \lambda(\mathbf{s},dt) = \lim_{|d\mathbf{s}|\to 0} \lambda(d\mathbf{s},t) \tag{1.63}$$

Combining the notation of (1.60) and (1.61) with the notation of (1.62) and (1.63), similar expressions for the average conditional spatial intensity, $\lambda(d\mathbf{s}|t)$, and the average conditional temporal intensity, $\lambda(dt|\mathbf{s})$, are obtained as:

$$\lambda(d\mathbf{s}|t=t^*) = \frac{E[N(d\mathbf{s},t^*)]}{|d\mathbf{s}|} \tag{1.64}$$

and

$$\lambda(dt|\mathbf{s} = \mathbf{s}^*) = \frac{E[N(\mathbf{s}^*, dt)]}{|dt|} \tag{1.65}$$

The first-order intensities defined in this section may not be appropriate for all spatiotemporal point patterns. For the earthquake process which is both orderly in space and time the concept of a conditional intensity at time t or location $\mathbf{s}$ has no meaning. However, conditional intensities for earthquake processes can be constructed on intervals in time or regions in space (Rathbun (1996)[52]). The conditional intensities on intervals in space and/or time are denoted above by the average conditional intensities.

1.8.6 Second-order intensities

The extension of $\lambda(\cdot)$ to include a temporal component leads, by analogy, to the general definition of the ***second-order spatiotemporal intensity*** $\lambda_2(\cdot)$ as:

$$\lambda_2(\mathbf{s}_{i1}, \mathbf{s}_{j2}, t_1, t_2) = \lim_{\substack{|A_{i1}|\to 0\\ |A_{j2}|\to 0}} \frac{E[N(A_{i1})N(A_j2)]}{|A_{i1}||A_{j2}|} \tag{1.66}$$

where $A_{i1} = d\mathbf{s}_{i1} \times dt_1$ is an infinitesimal cylinder containing the point $(\mathbf{s}_{i1}, t_1)$ and $A_{j2} = d\mathbf{s}_{j2} \times dt_2$ is an infinitesimal cylinder containing the point $(\mathbf{s}_{j2}, t_2)$.

The marginal second-order intensity removes either the spatial or temporal component by integrating over the spatial or temporal space.

Definition 1.38. The **marginal second-order spatial intensity** is defined by:

$$\lambda_2(\mathbf{s}_{i*}, \mathbf{s}_{j*}, \cdot, \cdot) = \int_T \int_T \lambda_2(\mathbf{s}_{ik}, \mathbf{s}_{jl}, t_1, t_2) dt_1 dt_2 \tag{1.67}$$

where the integration is over all times.

Definition 1.39. The **marginal second-order temporal intensity** is defined by:

$$\lambda_2(\cdot,\cdot,t_1,t_2) = \int_{A_1}\int_{A_2} \lambda_2(\mathbf{s}_1,\mathbf{s}_2,t_1,t_2)d\mathbf{s}_1 d\mathbf{s}_2 \tag{1.68}$$

where the integration is over the region A at time t.

Conditioning on time in (1.66) gives us the ***conditional second-order spatial intensity*** defined by:

$$\lambda_2(\mathbf{s}_{i*},\mathbf{s}_{j*}|t=t^*) = \lim_{\substack{|d\mathbf{s}_{i*}|\to 0\\ |d\mathbf{s}_{j*}|\to 0}} \frac{E[N(d\mathbf{s}_{i*},t^*)N(d\mathbf{s}_{j*},t^*)]}{|d\mathbf{s}_{i*}||d\mathbf{s}_{j*}|} \tag{1.69}$$

Similarly, conditioning on location in (1.66) yields the ***conditional second-order temporal intensity*** defined by:

$$\lambda_2(t_1,t_2|\mathbf{s}=\mathbf{s}^*) = \lim_{\substack{|dt_1|\to 0\\ |dt_2|\to 0}} \frac{E[N(\mathbf{s}^*,dt_1)N(\mathbf{s}^*,dt_2)]}{|dt_1||dt_2|} \tag{1.70}$$

As with the first-order conditional intensities, the second-order conditional intensities allow examination of one component, either spatial or temporal, while holding the other component fixed.

Using similar notation to (1.61) and (1.62) we define the ***average second-order spatial intensity*** as:

$$\lambda_2(\mathbf{s}_{i1},\mathbf{s}_{j2},dt_1,dt_2) = \lim_{\substack{|d\mathbf{s}_{i1}|\to 0\\ |d\mathbf{s}_{j2}|\to 0}} \frac{E[N(d\mathbf{s}_{i1},dt_1)N(d\mathbf{s}_{j2},dt_2)]}{|d\mathbf{s}_{i1}||d\mathbf{s}_{j2}||dt_1||dt_2|} \tag{1.71}$$

and the ***average second-order temporal intensity*** as:

$$\lambda_2(d\mathbf{s}_{i1},d\mathbf{s}_{j2},t_1,t_2) = \lim_{\substack{|dt_1|\to 0\\ |dt_2|\to 0}} \frac{E[N(d\mathbf{s}_{i1},dt_1)N(d\mathbf{s}_{j2},dt_2)]}{|d\mathbf{s}_{i1}||d\mathbf{s}_{j2}||dt_1||dt_2|} \tag{1.72}$$

Combining the definitions in (1.69) and (1.70) an expression for ***average conditional second-order spatial intensity*** is given by:

$$\lambda_2(d\mathbf{s}_{i*}, d\mathbf{s}_{j*}|t = t^*) = \frac{E[N(d\mathbf{s}_{i*}, t^*)N(d\mathbf{s}_{j*}, t^*)]}{|d\mathbf{s}_{i*}||d\mathbf{s}_{j*}|} \tag{1.73}$$

and an expression for ***average conditional second-order temporal intensity*** is given by:

$$\lambda_2(dt_1, dt_2|\mathbf{s} = \mathbf{s}^*) = \frac{E[N(\mathbf{s}^*, dt_1)N(\mathbf{s}^*, dt_2)]}{|dt_1||dt_2|} \tag{1.74}$$

Chapter 2

Spatial point processes for tumor growth. Cobweb algorithm[1]

Lévy theory provides a potential mathematical framework to model space and space-time stochastic processes. In addition, spatial point processes define stochastic models for random patterns of points in $\mathbb{R}^2$. These processes play a special role in stochastic geometry as the building blocks of more complicated random set models. In this paper we focus on a family of Lévy-based spatial Cox processes to model and predict tumor growth. We develop a procedure to simulate the growing of tumors. This algorithm can be used to study the evolution in time of any 2 and 3-dimensional geometrical forms such as cancer skin and all type of boundary evolution. We analyze real data and show that the procedure developed works fine and is useful for prediction purposes.

The evolution in time of some objects is the subject of study of many researchers worldwide. Special attention has been given to cancer, and a way to understand this disease is to know how it evolves over time.

---

[1] This chapter is based on the published paper: "*A geometric approach to cancer growth prediction based on Cox Processes*" by Vlad and Mateu (2014)[1]

We can find, in reading, many studies about mathematical modeling of tumors; see, for example, Bramson and Griffeath (1981)[55], Cressie (1991)[56], Qi et al. (1993)[57], Lee and Cowan (1994)[58], Kansal et al. (2000)[59], Barndorff-Nielsen and Schmiegel (2004)[60], Jónsdóttir and Jensen (2005)[61], Jónsdóttir and Jensen (2008)[62].

In Richardson (1973)[63], the growth object in the plane at time t is a random subset Y_t of $\mathbb{Z}^2$ consisting of the "infected sites", described also by a Markov process. An uninfected site is transferred to an infected site with a rate proportional to the number of infected nearest neighbors. It can be shown that if Y_0 consists of a single site, then Y_t/t has a non-random shape as $t \to \infty$.

Bramson and Griffeath (1981)[55] denote the set of sites occupied by cancer cells ξ_t^A, at time t and given that the original cancerous population (ξ_t^0) occupies $A \in S_0$, and the processes $(\xi_t^\alpha)_{t\geq 0}$ are Markov, they define for $\lambda > 0$, a jump (growth) rate as

$$\begin{aligned} &A \to A \cup \{x\} \ (x \notin A) \text{ at rate } \lambda |\{y \in A : \parallel y - x \parallel = 1\}|, \\ &A \to A - \{x\} \ (x \in A) \text{ at rate } 1 \end{aligned}$$

where $|\Lambda|$ is the cardinality of $\Lambda \in S_0$ and $\parallel x \parallel$ is the Euclidean distance from x to 0. This study is a extension of the stochastic model defined by Williams and Bjerknes (1972)[64] to accommodate expansion of cancer cells. That model treats each cell as normal and abnormal (cancerous) independently and assume that both are located in a planar lattice $\mathbb{Z}^2$. Starting with a single abnormal cell at the origin and from the hypothesis that abnormal cells reproduce faster that normal cells, they assume that with each cellular division, one daughter cell stays fixed while the other usurps the position of a neighbor.

A related growth model in continuous space has been discussed in Deijfen (2003)[65]. For planar objects, the model is constructed from a spatio-temporal Poisson point process on $\mathbb{R}^3$,

$$\Psi = \{(x_i, t_i)\}$$

The random growing object $Y_y \subset \mathbb{R}^2$ is a subset of

$$\bigcup_{\{i:t_i \leq t\}} B(x_i, r)$$

constructed such that Y_t is always connected. Here, $B(x, r)$ is a circular disc with center x and radius r. In this model, t_i is thought of as a time point of outburst and x_i as the location of the outburst in the tumor. A closely related discrete-time Markov growth model has been proposed by Cressie (1991)[56]. This model can be characterized as a sequence of Boolean models,

$$Y_{t+1} = \cup \{B(x_i, r) : x_i \in Y_t\}$$

where $\{x_i\}$ is a homogeneous Poisson point process in $\mathbb{R}^2$; see Cressie and Laslett (1987)[66], Cressie (1991)[56] and Cressie (1993)[18].

In all these papers it is obvious that the form of the cancer in the future depends by the edge and structure of the cancer in the past (function f) and also by some external factors like mitosis, nature of cancer (benign or malign), biological tissue density, etc (all these factors can be included in a function g). So, a *very general growth model* can be expressed as

$$Y_{t+1} = f(Y_t) + g \tag{2.1}$$

To avoid complicating the model we only consider the external edge of the tumor and do not deal with its internal structure. Then we assume that the object of study can be a shape with a particular boundary of whose genesis is a single point. Let us study this as a point process and let us make some geometrical interpretations to calculate the rate growth. For this it is not strictly necessary to have deterministic expressions of the functions f and g and make a complex model which aims to offer the values of the boundary object at any instant time. We can calculate these values with an algorithm by observing the speed of expansion of the tumor, expressed as a constant velocity of vectors in some directions.

The aim of this research is to observe the dynamics of cancer tumors and to develop and implement new methods and algorithms for prediction of tumor growth. We offer some tools to help physicians for a better understanding and treatment of this disease. Using a prediction method, and comparing with the real evolution a physician can note if the prescribed treatment has the desired effect, and according to this, if necessary, to take the decision of surgically intervention.

The plan of the chapter is the following. Section 2.1 presents a short overview of the theory of Lévy bases and integration with respect to such bases is given, we recall standard results about Spatial Cox processes, and finally we propose different types of growth models. In Section 2.2 a new algorithm, the Cobweb, is presented and developed based on the proposed methodology. The implementation in Matlab software comes in Section 2.3. Section 2.4 presents some real data analysis.

2.1 Spatial Cox point processes

A spatial point pattern is a set of points $\{\mathbf{x}_i \in A : i = 1, \ldots, n\}$ for some planar region A. Often A is a sampling window within a much larger region and it is reasonable to regard the point pattern as a partial realization of a stochastic planar point process, the events consisting of all points of the process which lie within A.

Let N be a spatial point process that is defined on $\mathbb{R}^2$ but is observed on a finite observation window W. For an arbitrary Borel set $A \in \mathbb{R}^2$, let $|A|$ and $N(A)$ denote the area of A and the number of events from N that are in A, respectively. In some applications, it is reasonable to think of the spatially varying intensity function, $\lambda(\mathbf{x})$ of a point process to be itself a realization of an underlying stochastic process $\Lambda(\mathbf{x})$.

A point process X is a *Cox process* if: (a) $\Lambda(\mathbf{x})$ is a non-negative valued stochastic process; and (b) conditional on the realization of $\Lambda(\mathbf{x})$, the point process is an inhomogeneous Poisson process with intensity function $\Lambda(\mathbf{x})$ Cox (1994)[5]. In this case, we say that X is a Cox process driven by Λ. In this context, the resulting point process inherits the properties of the $\Lambda(\mathbf{x})$ process in a natural way.

Cox processes provide useful and frequently applied models for aggregated spatial point patterns where the aggregation is due to a stochastic heterogeneity. Indeed, Λ usually models this unobserved random heterogeneity. Shot noise Cox processes, log Gaussian Cox processes and log shot noise Cox processes will appear as natural building blocks in a modeling framework for Cox processes.

2.1.1 Lévy-based Cox processes

Let $(\Omega, \mathcal{A})$ be a measurable space. We assume that Ω is a Borel subset of $\mathbb{R}^d$, and $\mathcal{A}$ is the δ-ring $\mathcal{B}_b(\Omega)$ of bounded Borel subsets of Ω. We consider a collection of real-valued random variables $L = \{L(A), A \in \mathcal{A}\}$ with the following properties:

- $L(A_1), \ldots, L(A_n), \ldots$ are independent random variables for every sequence $\{A_n\}$ of disjoint sets in $\mathcal{A}$, and $L(\cup_n A_n) = \sum_n L(A_n)$ a.s. provided $\cup_n A_n \in \mathcal{B}_b(\Omega)$,
- for every A in $\mathcal{A}$, $L(A)$ is infinitely divisible.

If L has these properties, L is called a Lévy basis. In addition, L is a non-negative Lévy basis if $L(A) \geq 0$ for all $A \in \mathcal{A}$ (cf. Barndorff-Nielsen and Schmiegel (2004)[60]).

For a random variable X, Jónsdóttir and Jensen (2008)[62] denote the cumulant function $\log \mathbb{E}(e^{i\,\nu\,X})$ by $C(\nu, X)$. If L is a Lévy basis, then the cumulant function of $L(A)$ is expressed as

$$C(\nu, L(A)) = i\nu a(A) - \frac{1}{2}\nu^2 b(A) + \int_{\mathbb{R}} \left(e^{i\nu r} - 1 - i\nu r \mathbf{1}_{[-1,1]}(r)\right) U(dr, A), \tag{2.2}$$

where a is a σ-additive set function on $\mathcal{A}$, b is a measure on $\mathcal{A}$, $U(dr, A)$ is a measure on $\mathcal{A}$ for fixed dr and a Lévy measure on $\mathcal{B}(\mathbb{R})$ for each fixed $A \in \mathcal{A}$ (i.e. $U(\{0\}, A) = 0$ and $\int_{\mathbb{R}}(1 \wedge r^2) U(dr, A) < \infty$, where $\wedge$ denotes minimum).

The measure U is referred to as the *generalised Lévy measure* and L is said to have the *characteristic triplet* (a, b, U). In addition, (a) if $b = 0$ then L is called a *Lévy jump basis*, and (b) if $U = 0$ then L is a *Gaussian basis*. A general Lévy basis L can always be written as a sum of a Gaussian basis and an independent Lévy jump basis.

A particular example of a Gaussian Lévy basis is obtained by attaching independent Gaussian random variables $\{X_i\}$ to a locally finite sequence $\{\eta_i\}$ of fixed points and defining

$$L(A) = \sum_{\eta_i \in A} X_i\ , \quad A \in \mathcal{A}.$$

Let $\mathcal{S}$ be a Borel subset of $\mathbb{R}^d$. A point process X on $\mathcal{S}$ is called a *Lévy driven Cox Process* (LCP) if X is a Cox process with a driving field

$$\Lambda(\xi) = \int_{\Omega} k(\xi, \eta) L(d\eta), \quad \xi \in \mathcal{S}, \tag{2.3}$$

where L is a non-negative Lévy basis on Ω.

Furthermore, k is a non-negative function on $\mathcal{S} \times \Omega$ such that $k(\xi, \cdot)$ is integrable with respect to L for each $\xi \in \mathcal{S}$ and $k(\cdot, \eta)$ is integrable with respect to the Lebesgue measure on $\mathcal{S}$ for each $\eta \in \Omega$.

Note that it is always possible for each pair (k, L) to construct an associated pair $(\bar{k}, \bar{L})$ generating the same driving field Λ where now $\bar{k}(\cdot, \eta)$ is a probability kernel. We may simply let

$$\begin{aligned} \bar{k}(\xi, \eta) &= k(\xi, \eta)/\alpha(\eta), \\ \bar{L}(d\eta) &= \alpha(\eta) L(d\eta) \end{aligned}$$

where

$$\alpha(\eta) = \int_{\mathcal{S}} k(\xi, \eta) d\xi.$$

It is important to note that from the non-negativity of the Lévy basis L, we get that L is equivalent to a random measure on Ω. Thus, the measurability of Λ follows from the measurability of k as a function of η and ξ. Therefore, Λ is a well-defined random field and (under the condition of local integrability) the driving measure $\int_B \Lambda(\xi) d\xi$, $B \in \mathcal{B}_b(\mathcal{S})$, is also a well-defined random measure determined by the finite-dimensional distributions of L.

The function k and the Lévy basis L will be chosen such that Λ is almost surely locally integrable, i.e. $\int_B \Lambda(\xi) d\xi < \infty$ with probability 1 for $B \in \mathcal{B}_b(\mathcal{S})$. A sufficient condition for the last property is that, (cf. Møller (2003)[67], Remark 5.1)

$$\int_B \mathbb{E}\Lambda(\xi) d\xi < \infty, \quad B \in \mathcal{B}_b(\mathcal{S}) \tag{2.4}$$

2.1.2 Lévy-based tumor growth modeling

Let us denote the growing object as a planar object at time t by $Y_t \in \mathbb{R}^2$, and assume that Y_t is compact and *star-shaped* with respect to a point $O(x_0, y_0) \in Y_t$ for all t. We treat here a star-shaped object like a two-dimensional geometric shape and the growth model like a rigid transformation in time of the primary star-shaped object, defining a second star-shaped object that includes the boundary of the initial object.

In geometry, two subsets of a Euclidean space have the same shape if one can be transformed into the other by a combination of translations, rotations (together also called rigid transformations), and uniform scalings. Note that we talk about star-shaped that can grow its boundary in a random way. So we can not say that the transformation of the initial star-shape object in time is a rigid transformation.

The boundary of Y_t can be determined from the variation in time t and direction (from angle) $\phi \in [0, 2\pi)$ of the vector $\vec{R}_t(\phi)$ denoted by the radial function,

$$R_t(\phi) = \max\{r : (x_0, y_0) + r(cos\phi, sin\phi) \in Y_t\}; \quad \phi \in [0, 2\pi) \qquad (2.5)$$

In Figure 2.1 we show an example of such star-shape object noted with Y_t and $R_t(\phi)$ is the distance from a reference point $O(x_0, y_0)$ to the boundary of the object.

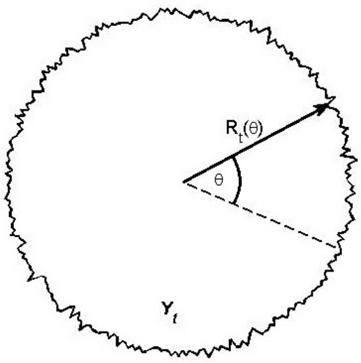


FIGURE 2.1: The star-shape object Y_t.

Jónsdóttir and Jensen (2008)[62] consider this as a random variable $X_t(\sigma)$ depending on time t and position σ in space. They assume that $(\sigma, t) \in \Omega = \mathcal{S} \times \mathbb{R}$, where $\mathcal{S} \subseteq \mathbb{R}^n$. A Lévy-based spatio-temporal model for $X = \{X_t(\sigma) : (\sigma, t) \in \Omega\}$ is based on the ambit set $A_t(\sigma)$ associated with each point $(\sigma, t) \in \Omega$, which defines the dependency on the past at time t and position σ, and satisfies the conditions

$$\begin{gathered}(\sigma, t) \in A_t(\sigma) \\ A_t(\sigma) \subseteq \mathbb{S} \times (-\infty, t]\end{gathered}$$

The *linear spatio-temporal Lévy model* for $X = \{X_t(\sigma) : (\sigma, t) \in \Omega\}$ is defined as

$$X_t(\sigma) = \int_{A_t(\sigma)} f_t(\xi, \sigma) L(d\xi) \tag{2.6}$$

where L is a Lévy basis and $f_t(\xi, \sigma)$ is the deterministic weight function. The process

$$\bar{X} = \{\exp\left(X_t(\sigma)\right) : (\sigma, t) \in \Omega\}$$

is said to follow an exponential spatio-temporal Lévy model.

So, the *model that describes the growth of a planar star-shaped object*, using its radial function $R_t(\phi)$ at time t and angle ϕ, can start from the time derivative of the radial function equation

$$\frac{\partial}{\partial t} R_t(\phi) = \mu_t(\phi) + \int_{A_t(\phi)} f_t(\xi; \phi) L(d\xi); \quad \phi \in [0, 2\pi) \tag{2.7}$$

which is the growth rate (cf Jónsdóttir and Jensen (2008)[62]). Here, L is the Lévy-basis on $[0, 2\pi) \times \mathbb{R}$; $A_t(\phi) \subseteq [0, 2\pi) \times (-\infty, t]$ is a subset of the past of time t, called *ambit set* (Barndorff-Nielsen and Schmiegel (2004)[60]); $f_t(\cdot, \phi) : [0, 2\pi) \times \mathbb{R} \to \mathbb{R}$ is a deterministic weight function (assumed to be suitable for the integral to exist) and the deterministic function $\mu_t : [0, 2\pi) \to \mathbb{R}$ contributes to the overall growth pattern while the stochastic integral determines the dependence structure in the growth process.

The ambit set $A_t(\phi)$ plays an important role in this modeling approach and affects the degree of dependence on the past. The extent of the dependence on the past may be specified by the minimal time-lag $T(t)$ such that

$$A_t(\phi) \subseteq [0, 2\pi) \times [t - T(t),\, t]; \quad \phi \in [0, 2\pi)$$

The form of the ambit set $A_t(\phi)$ will depend on the specific growth process being modeled. For the interpretation of (2.7) as a growth model, Jónsdóttir and Jensen (2008)[62] represent the ambit set as a stochastic subset of the growing object. This is possible if the stochastic time transformation $t \to R_t(\phi)$ is non-decreasing for each $\phi \in [0, 2\pi)$. They represent the ambit set $A_t(\phi)$ as a subset of Y_t,

$$\bar{A}_t(\phi) = \Big\{ \big(R_s(\theta)\cos\theta, R_s(\theta)\sin\theta\big) : (\theta, s) \in A_t(\phi) \Big\} \tag{2.8}$$

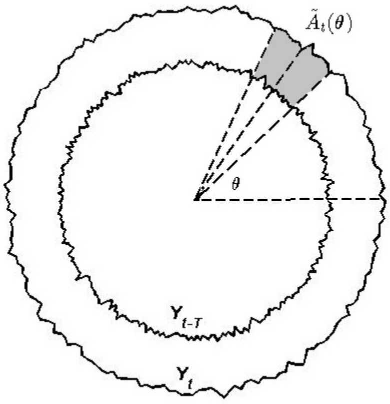


Figure 2.2: Stochastic representation of $\bar{A}_t(\phi)$.

It follows from the fact that $A_t(\phi) \subseteq [0, 2\pi) \times (-\infty, t]$ that $\bar{A}_t(\phi)$ is actually a subset of Y_t (see Figure 2.2).

Furthermore, since $(\phi, t) \in A_t(\phi)$, the set $\bar{A}_t(\phi)$ touches the boundary of Y_t at the point $\big(R_t(\phi)\cos\phi, R_t(\phi)\sin\phi\big)$. It is the "events" in $\bar{A}_t(\phi)$ that influence the growth rate at time t in direction ϕ.

In the particular case where L is a *Poisson basis and* Ψ *the associated Poisson point process on* $[0, 2\pi) \times \mathbb{R}$, then *the growth model* can be written as

$$\frac{\partial}{\partial t}R_t(\phi) = \mu_t(\phi) + \sum_{\bar{\xi}\in\bar{\Psi}_t\cap\bar{A}_t(\phi)} \bar{f}_t(\bar{\xi};\phi) \tag{2.9}$$

where the parameter $\bar{\Psi}_t$ of the sum is a subset of Y_t

$$\bar{\Psi}_t = \Big\{\big(R_{t_i}(\theta_i)\cos\theta_i, R_{t_i}(\theta_i)\sin\theta_i\big)\,:\, t_i \le t\Big\} \tag{2.10}$$

Finally, if

$$\bar{f}_t\big((s\cos\theta, s\sin\theta);\,\phi\big) = f_t\big((\theta,s);\phi\big) \tag{2.11}$$

and according to (2.9), the growth rate at time t in the direction ϕ depends on the outbursts at time points before t which lie in the stochastic neighborhood $\bar{A}_t(\phi)$. Under (2.7), the induced model for $R_t(\phi)$ will be (cf Jónsdóttir and Jensen (2008)[62]) of the same linear form, since

$$\begin{aligned} R_t(\phi) &= R_0(\phi) + \bar{\mu}_t(\phi) + \int_0^t\int_{A_s} f_s(\xi;\phi)L(d\xi)ds = \\ &= R_0(\phi) + \bar{\mu}_t(\phi) + \int_{\bar{A}_t} \bar{f}_t(\xi;\phi)L(d\xi) \end{aligned} \tag{2.12}$$

where R_0 is the radial function at time $t = 0$, and

$$\bar{\mu}_t(\phi) = \int_0^t \mu_s(\phi)ds \tag{2.13}$$

$$\bar{A}_t(\phi) = \bigcup_{0\le s\le t} A_s(\phi) \tag{2.14}$$

$$\bar{f}_t(\xi;\phi) = \int_0^t \mathbf{1}_{A_s(\phi)}\,(\xi)f_s(\xi;\phi)ds \tag{2.15}$$

Note that the ambit sets associated with the radial function itself are increasing, that is,

$$t_1 \le t_2 \Rightarrow \bar{A}_{t_1}(\phi) \subseteq \bar{A}_{t_2}(\phi)$$

Another model proposed by Jónsdóttir and Jensen (2008)[62] is expressed in terms of the time derivative of $\ln\big(R_t(\phi)\big)$

$$\frac{\partial}{\partial t}\Big(\ln\big(R_t(\phi)\big)\Big) = \mu_t(\phi) + \int_{A_t(\phi)} f_t(\xi;\phi)L(d\xi) \tag{2.16}$$

and *the induced model is an exponential spatio-temporal Lévy model*

$$R_t(\phi) = R_0(\phi)\ \exp\left(\bar{\mu}_t(\phi) + \int_{\bar{A}_t(\phi)} \bar{f}_t(\xi;\phi)L(d\xi)\right) \tag{2.17}$$

The choices of Lévy basis L, ambit sets $A_t(\phi)$, weight functions $f_t(\xi;\phi)$ and $\mu_t(\phi)$ completely determine the growth dynamics. These four ingredients can be chosen arbitrarily and independently, which results in a great variety of different growth dynamics.

Finally, Jónsdóttir and Jensen (2005)[61] propose a *Gaussian radial model for star-shaped objects.* The object at time $t+1$ is a stochastic transformation of the object at time t such that the radius vector function of the object fulfils

$$R_{t+1}(\phi) = R_t(\phi) + Z_t(\phi), \phi \in [0, 2\pi) \tag{2.18}$$

where Z_t is a cyclical Gaussian process

$$Z_t(\phi) = \mu_t + \Sigma_{k=1}^{\infty}[A_{t,k}\cos(k\phi) + B_{t,k}\sin(k\phi)]$$

and assumed that the coefficients μ_t, $A_{t,k}$ and $B_{t,k}$ are the coefficients of the Fourier series of Z_t which has an important geometric interpretations relating to the growth process.

In the next section we offer a geometric interpretation of the growth tumor determined by the radius R_t and angle ϕ values at different instants of time t and we propose new algorithm to calculate the growing rate.

2.2 Modeling tumor growth: a new algorithm

Principal questions for this subject are how fast the tumor grows, how rapidly does it invade and replace brain tissue, and what is the life expectancy of the patient? In general, tumor growth depends on the rate of mitosis (birth of new cells) and the rate of apoptosis (cell death). A tumor in which the rate of mitosis is equal to the rate of apoptosis does not appear to grow - it stays the same size as new tumor cells and simply replaces cells which die and the number of tumor cells stays the same.

To make a prediction about tumor growth we first need data (at least two images) at a predetermined interval time, to see if the tumor is growing and what is the relative velocity. Once we have located the tumor, the next step is to make a second tomography and compare the boundary of the tumor. In Figure 2.3 we show a tomography of a tumor and its location within the brain.

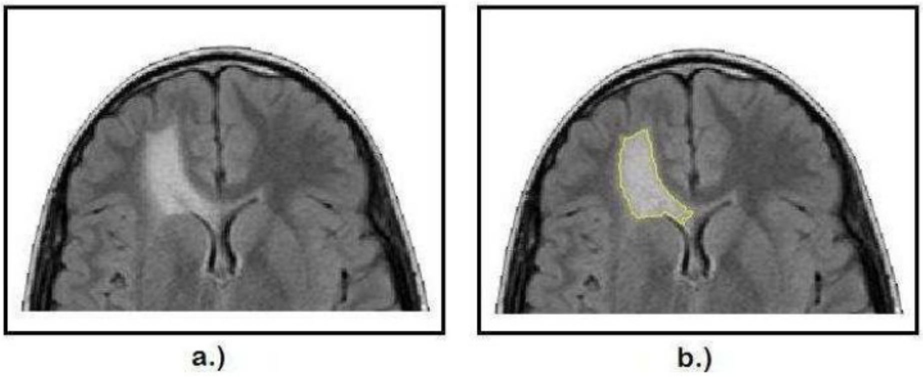


Figure 2.3: Brain tumor (a) and its location (b)

The tomography should be done in the same conditions as the previous image. The time period between two consecutive analysis is set as a parameter resolution k. Prediction of tumor growth directly depends on a good image acquisition.

In Figure 2.4 we have the second image after one month from when the disease was discovered. With the red line the tumor is marked

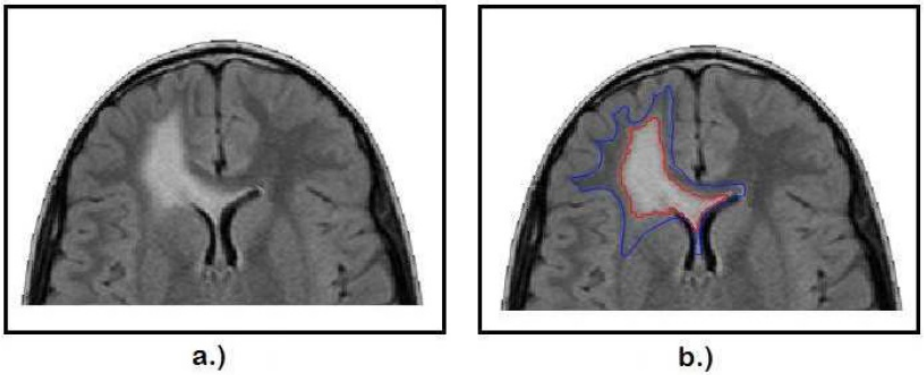


FIGURE 2.4: Second image acquisition after one month: a) original image acquisition; b) boundary of tumor (red) within the sample space (blue)

and in the right-hand picture we consider a sample space like a "star-shape" delimited by the blue line.

The sample space must be as large as possible, in the worst case it could be the entire cranial box. But we can also assume that in a period of time, the tumor can not grow over certain limits which depend on the structure of the brain.

We can easily construct a sample space which includes the boundary of the future tumor. The form of this sample space is directly influenced by the shape and positioning of the brain bulbs. It is a fact that the tumors grow more easily in some directions which depend on the density and the nature of nearby biological material. Now we can compare the images to see the evolution in time of the tumor. As we can see the tumor tends to grow more in certain directions which can be defined with vectors of growth of the tumor.

In Figure 2.5 we note the discovered tumor at time t with the yellow boundary, the growth tumor at time $t + \Delta T$ ($\Delta T = 30$ days) with red boundary, and the star-shape representing the sample space with a blue line. In the right-hand side we show in black lines the vectors from the center of tumor to the limits of the sample space.

The intersections of these vectors with each boundary give the value of the vector at time t, respectively $t + \Delta T$. ΔT represents the time

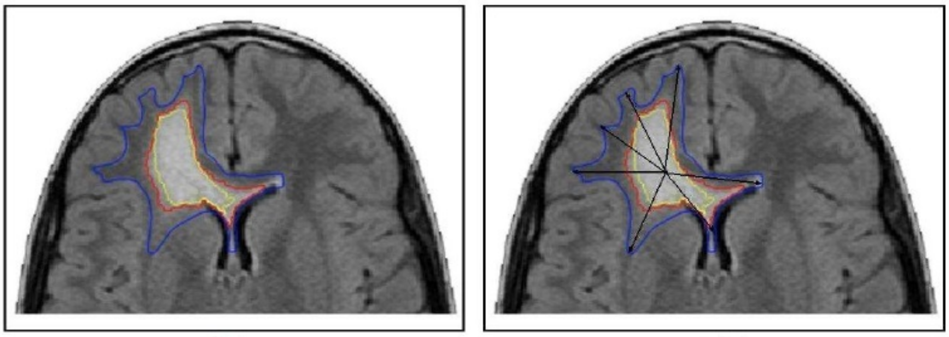


FIGURE 2.5: Superimposed images of tumor

period between analysis (here one month).

In Figure 2.6 we graphically explain how we use the vectors $\vec{R}_t^j$ to predict the growth of the tumor. The direction of each vector $\vec{R}_t^j$, $j = 1, \ldots, n$ is given by the line from the center $O(x_0, y_0)$ to each n inflexion of the star-shape (in this case we consider only $n = 7$). We denote the starting point time when we make the first tomography with t_0 as the initial time for our computation.

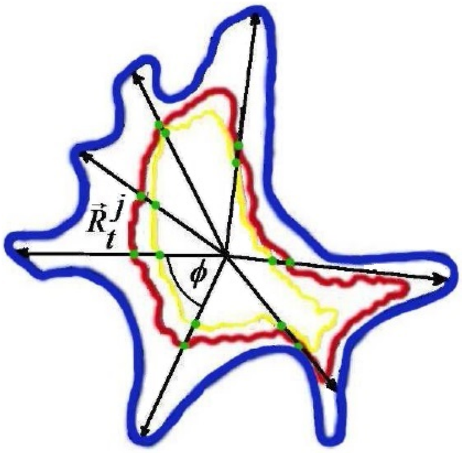


FIGURE 2.6: Tumor growth in star-shape

The area of the initial tumor A_{t_0} for a time t_0 is given by the boundary of the Y_{t_0} (yellow line in Figure 2.6), and the area of the tumor $A_{t_0+\Delta T}$ at time $t_0 + \Delta T$ (after one month) is given by the boundary of $Y_{t_0+\Delta T}$ (red line in Figure 2.6).

Suppose that the velocity of growth is constant in time, then the value of the vector $\vec{R}_{t_i}^j$ at the moment $t_i = t_0 + \alpha\Delta T$, represented by the radius r_i^j, is given by

$$r_i^j = \| \vec{R}^j_{t_0+\alpha\Delta T} \| = \| \vec{R}^j_{t_{i-1}+\Delta T} \| + k^{-1} \left(\| \vec{R}^j_{t_0+\Delta T} \| - \| \vec{R}^j_{t_0} \| \right), \tag{2.19}$$

for $i = 1, \ldots, k$ and $j = 1, \ldots, n$.

The parameter α is the period of time whereupon we wish to make the estimation and this is given by the linear resolution

$$k = \frac{\Delta T}{\alpha} \tag{2.20}$$

The angular resolution represents the circle divided by the number of vectors, which in cylindrical coordinates, means

$$n = \frac{2\pi}{\phi} \tag{2.21}$$

In equation (2.19) the final term of the right-hand side, is a constant defining the "length step"

$$l_k = k^{-1} \left(\| \vec{R}^j_{t_0+\Delta T} \| - \| \vec{R}^j_{t_0} \| \right)$$

and if we write $\| \vec{R}^j_{t_{i-1}+\Delta T} \| = r^j_{i-1}$ then (2.19) becomes

$$r_i^j = r_{i-1}^j + l_k \tag{2.22}$$

We use the cylindrical coordinates to calculate the area of $\bar{A}_t(\phi)$ (see Figure 2.2) which help to calculate the predicted area $\bar{A}_{t+\Delta T}(\phi)$ of the tumor after a period of time. To calculate this area we make a discretization of the area region $\bar{A}_t(\phi)$, in n angles ϕ_j, $j = 1, \ldots, n$ and k radius r_i, $i = 1, \ldots, k$. So, we split this region into $n \times k$ surfaces.

In Figure 2.7, the gray region represents the jth rate of growing tumor in time ΔT, noted here as $\bar{A}^j_{t_0+\Delta T}(\phi) - \bar{A}^j_{t_0}(\phi)$ with $j = 1, \ldots, n$

(n = angular resolution). In this case, the linear resolution is $k = 1$.

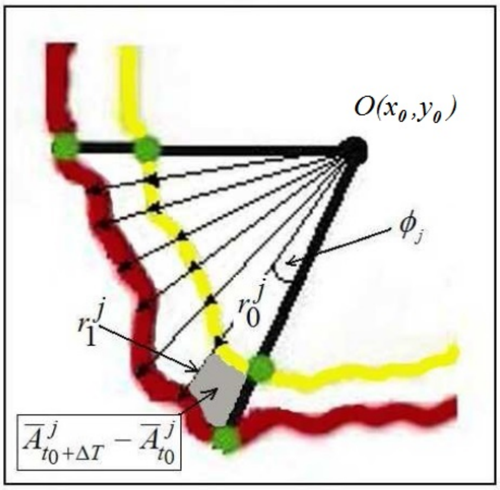


FIGURE 2.7: Calculation of the growth tumor at time $t + \Delta T$

Therefore the jth portion of area at time t_0 is

$$\bar{A}^j_{t_0}(\phi) = \int_{\phi_j} R^j_{t_0}\big(sin(\phi_j) + \cos(\phi_j)\big)d\phi_j \tag{2.23}$$

and the jth portion of area at time $t_0 + \Delta T$ is

$$\bar{A}^j_{t_0+\Delta T}(\phi) = \int_{\phi_j} R^j_{t_0+\Delta T}\big(sin(\phi_j) + \cos(\phi_j)\big)d\phi_j \tag{2.24}$$

Now we have the area of the initial tumor

$$\bar{A}_{t_0}(\phi) = \varepsilon + \sum_{j=1}^{n} \bar{A}^j_{t_0}(\phi) \tag{2.25}$$

and the area of the tumor from the second tomography is given by

$$\bar{A}_{t_0+\Delta T}(\phi) = \varepsilon + \sum_{j=1}^{n} \bar{A}^j_{t_0+\Delta T}(\phi) \tag{2.26}$$

The area of the growth tumor after a time period is composed from $n \times k$ elementary areas $\bar{A}^j_i(\phi_j)$, and is given by

$$A_{t_0+\alpha\Delta T} = \varepsilon + \sum_{i=1}^{k} \bar{A}^j_{t_i}(\phi) \tag{2.27}$$

where $\bar{A}^j_{t_i}(\phi)$ is the jth predicted portion of area at time $t_i + \alpha\Delta T$, given by

$$\bar{A}^j_{t_i}(\phi) = \varepsilon + \sum_{j=1}^{n} \bar{A}^j_i(\phi) \tag{2.28}$$

and $\bar{A}^j_i(\phi)$ is the ith and jth predicted portion of area

$$\bar{A}^j_i(\phi_j) = \int_{\phi_j} r^j_i \big(\sin(\phi_j) + \cos(\phi_j)\big) d\phi_j \tag{2.29}$$

Let us now formulate our proposed algorithm. We name it *Cobweb* (see below in step 3 for further explanations). We need to choose the number of vectors (angular resolution) and decide for the period of prediction (linear resolution). The precision of the prediction depends on these choices. Once we have this and the values of the vectors at time t_0 and $t_0 + \Delta T$ we can start the procedure:

Step 1 We compute the value of vector $\vec{R}^j_{t_{i+1}} = r^j_{i+1}$ using the current value of this vector (at time t_i) and we add the difference of the value of this vector at the current time with the value of the immediate past time, with expression (2.22). This is represented by a point of the future bounded tumor in the direction of this vector. The union of all these points gives the entire tumor.

Step 2 We calculate the area $\bar{A}^j_i(\phi)$ with expression (2.29).

Step 3 The general formula for computing the growing rate after a period t_i of time is given from the fact that a portion of area calculated in the step before is used in the current step, and in turn it is used on the following step (like a spider who is building its cobweb)

$$A^j_{i+1} = \varepsilon + \sum_{i=1}^{k}\sum_{j=1}^{n}\left(\bar{A}^j_i + n^{-1}k^{-1}\left(\bar{A}^j_i - \bar{A}^j_{i-1}\right)\right) \tag{2.30}$$

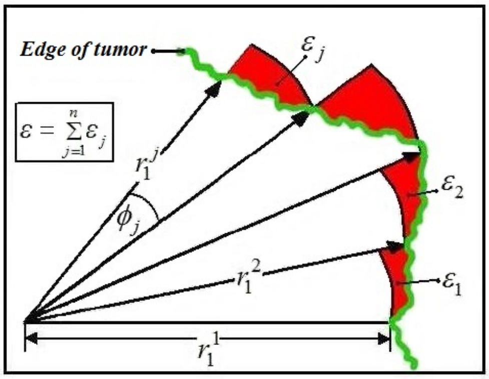


FIGURE 2.8: Error propagation

The error ε (red region in Figure 2.8) is given by the fact that we compute an approximated area with this method. This error can be diminished using a numerical method such as least squares, the trapezium method or just by increasing the resolution.

In Figure 2.9 we provide an estimation of the growth tumor after two months and the result comes in the green region. In this case the time period is $2\Delta T$. This algorithm can be extended to a three-dimensional space by replacing the vector $\vec{R}^j_t$ with a surface generated from this vector with spherical coordinates. In this case we should define a new resolution (radial resolution).

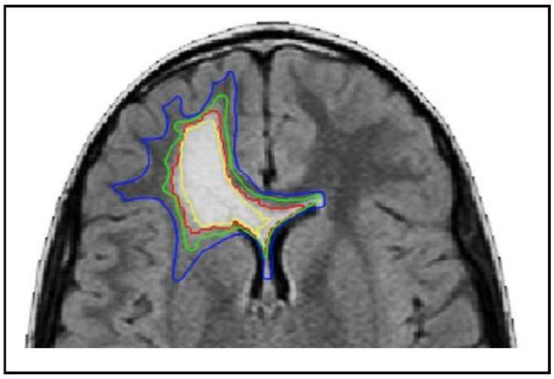


FIGURE 2.9: Predicted tumor after two months

2.3 Software

We implemented a set of new functions in Matlab software to proceed with our approach based on images coming from magnetic resonance imaging (MRI), computed tomography (CT) or another such techniques.

2.3.1 Input data

As input data we need at least two images (ideally three) of the same brain tumor, taken at predetermined or known time intervals. The precision of our method of prediction is given mainly by the number of vectors in which the direction of the tumor growth is forecast; they divide the circle counterclockwise in a number of angles equal to the number of vectors, so that we can say that they represent the angular resolution. We also need to input the time between any two images, and the elapsed time since the second image.

In our case, these last two input data are days but can also be months or hours (entire unit time). The accuracy of prediction also depends on these elections based on longitudinal resolution, which gives the unit time of growth rate per vector.

2.3.2 Procedures that must be fulfilled

The main objective of this code is to implement the algorithm presented in the previous section, and obtain a prediction of the tumor boundary. To perform this, we need to follow some stages:
(a) Comparison of the two (or three) images taken as input data to determine the rate of increase or decrease of the tumor from each chosen vector (direction);
(b) To obtain the most accurate growth rate in each direction it is

absolutely necessary that the images are taken under the same conditions, observing a single point of reference, respecting and preserving the same cartesian reference for all future tests of the same patient;
(c) In a two-dimensional situation we have to determine the approximate tumor center for the first image, and it will be preserved for all other images (second and/or third image if necessary);
(d) A good precision is obtained if the images to be compared are taken on the same plane and do overlap. If this is not the case, this means that the patient does not have the same position and then we have to apply some transformations (translations and/or rotations);
(e) To get a more precise outline of the tumor prediction, we need to compare the different stages of the tumor development. This will be done by entering the coordinate points of the contour for each instance of time. By default the number of points is set to 20 but the user can modify this to increase the precision in detriment of computing time. The user must choose the coordinates of designated points (number of contour points) exactly at the intersection of the line vector with the contour line of the edge of the tumor (see Figure 2.2 and Figure 2.7);

$$P_i(x_i; y_i) = \overrightarrow{R_t^j} \bigcap Y_t$$

(f) Besides the angular resolution (the number of vectors that will calculate the prediction) accuracy of calculation will be influenced by how precise and accurate the contour points are chosen. To ease the task we use the image segmentation function (*"imcontour"* in Matlab [68]) that will depict more clearly the outline of the tumor;
(g) We can interpolate the outline by using splines over the existing points. to get a more precise outline.

The output of these functions represent the contour of the predicted tumor after the time designated by the user. This will be plotted in the same two-dimensional plane with the last tumor, together with all stages of the tumor development in time.

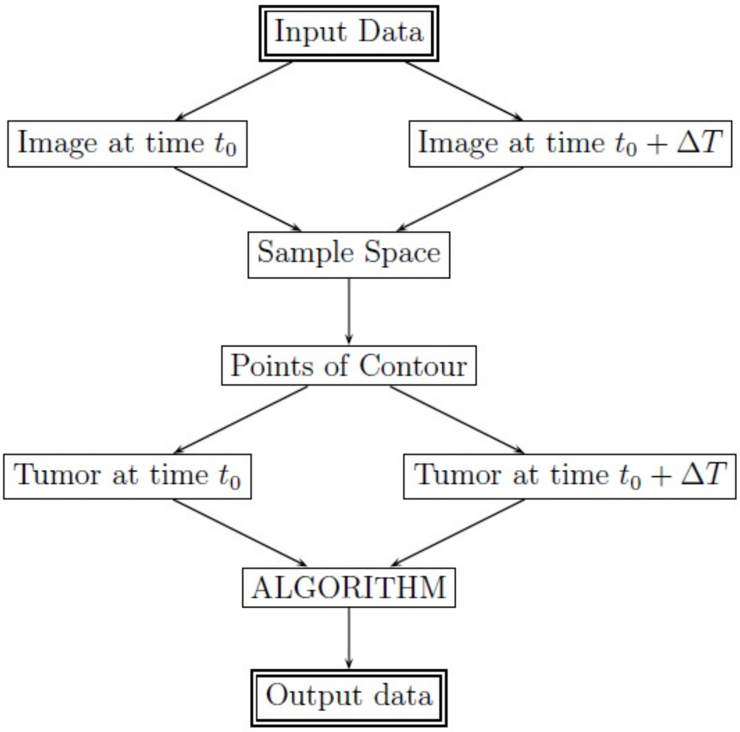


FIGURE 2.10: Diagram of the script

In Figure 2.10 we show a diagram of the built software.

2.4 Real data analysis

We study a particular tumor located in the Central Nervous System and called *glioblastoma multiform*. In conformity with the World Health Organization, this tumor is the most aggressive tumor with type and grade according to the IV-th classification. Observing the scaned images, it is clear that there is presence of multiple tumors in the body, a fact called metastasis. These metastatic tumors are children of primary tumors from breast, lung, colon, stomach and skin (melanoma), but in our case, the first one was the brain tumor.

A patient with glioblastoma multiform has an average life span of one year, receiving radiation therapy, steroids, and anticonvulsants.

Otherwise the patient dies long before one year. For the patients affected by this type of tumor, a neurological deterioration is noted producing difficulty in organizing and coherently expressing ideas, and then losing the mobility function, all depending on the order in which the tumor affects the brain and areas focused on memory, speech, motor function, etc.

Here we selected three images taken in the same plane: two from 2009, November the 9th, and December the 8th, and one from 2010, in January the 10th. Using the first two images we can make a prediction of the tumor growth for the next temporal instant, and we can thus compare the prediction with the original third image.

Using the function "*linie.m*"(Matlab[68]), we can determine the approximate center point of the tumor by choosing 4 points in the contour of the tumor, and opposite two by two. This will plot two lines, whose intersection provides the coordinate of the center point. We then determine the sample space (blue contour) and the growing directions (vectors, black lines). By default the angular resolution is 20, meaning that the user should choose 20 points to design the "Sample Space".

The choice of these points must take into account physical considerations such as:
(a) the coordinates of each chosen point can be at the limit of the tumor after a certain time but not too far away;
(b) if the tumor grows very quickly in one direction and the coordinate value that it can take in that direction is not physically possible (for example, it can get out of the head box) then, the prediction takes as value the limit point of the sample space, which represents the maximum allowed values in this direction;
(c) the designation of the sample space points can be seen as an outline of the tumor when the time is very large;

(d) the last point in the sample space must have the same coordinates as the first point, and if the user misses this, the Matlab program will do this automatically.

The sample space must include the tumor contour at time t and $t + \Delta T$ with which we can make the prediction. The user must select the same number of points, but this time, at the intersection of the vectors (black lines) with the boundary of the tumor. This action will draw the outline of the first tumor with a yellow line (see Figure 2.11).

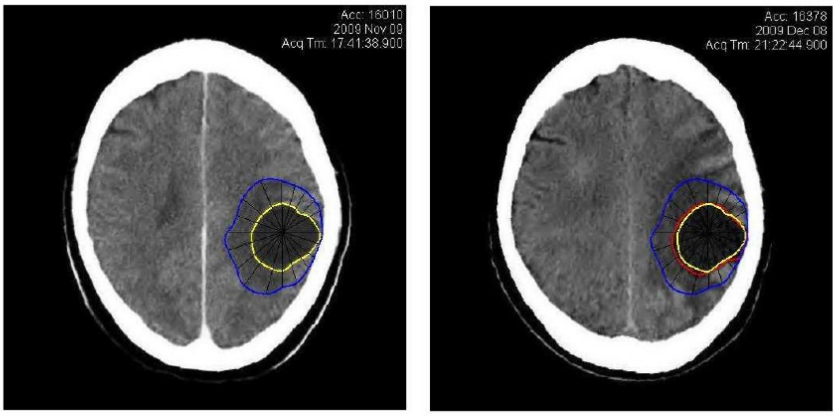


FIGURE 2.11: Real data analysis. A yellow line represents the tumor at time t (time when it was discovered), and a red line represents the tumor after time $t + \Delta T$.

Then, we upload the next magnetic resonance image (the second analysis) and follow the same procedure to enter the contour points for the second tumor. This set of points should belong to the vectors and also to the boundary of the tumor, and so each point P_i must be chosen at the intersection of each vector $R^j_{t+\Delta t}$ with the boundary tumor $Y_{t+\Delta T}$. This will draw a red spline curve corresponding to the boundary of the tumor at time $t + \Delta T$ (see Figure 2.11).

In the next step, we apply our algorithm to predict the growing tumor after 33 days. The prediction (green line in Figure 2.12) will be drawn and calculated as a spline curve. If we have a third image that represents the tumor stage at time $t + \Delta T + \alpha$, we can directly compare the predictive results with the original ones.

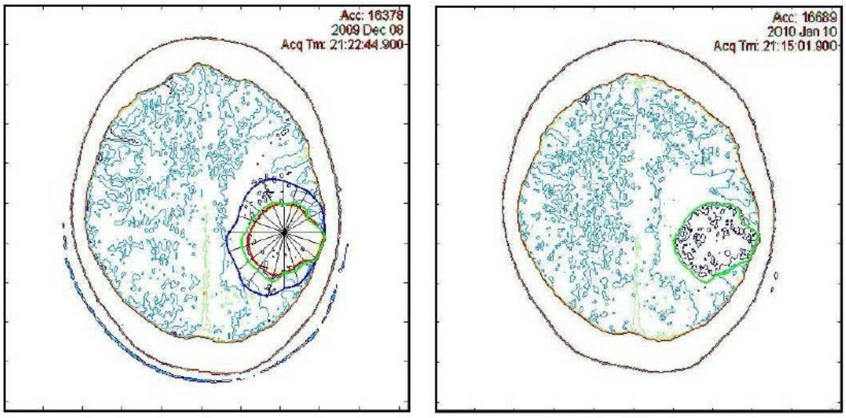


FIGURE 2.12: Predicted tumor

In Figure 2.12 we can see the result of the prediction. We use a segmented picture for a better observation of the contour of the tumor. The boundary of the predicted tumor is plotted in green on the background image from December. We can note the evolution of the tumor in time: the first stage comes in a yellow line, and second stage in a red line. In the right-hand side window we show the predicted tumor over the background image of the tumor from 10 of January.

2.5 Conclusions

The double stochastic process theory offers a mathematical background to study some natural and physical phenomena in the real world and it takes some conclusions and supplementary information about understanding what is happening in these complex systems.

Mathematical modeling always tries to find a compromise between simplicity of analysis and requirements of realism. On the one hand, we have extremely complex natural and biological systems; on the other hand, we need to formally address some quantitative issues about these systems which can be often done only through the use of mathematical models that may rest on grossly over-simplified assumptions.

On some occasions, a particular mathematical formalism seems to be pre-adapted to a variety of natural and biological systems and can be profitably used to model a diverse set of processes. Double stochastic Cox processes are one class of such models, used here to solve real problems in the field of medicine.

For most of the realistic problems, the solution of the corresponding exact equation is in practice impossible, so we need to make approximations. Making approximations to solve difficult problems is not a new idea. Appropriate models enable accurate prediction of future behavior, which can be used to control and optimize various aspects of the system in question. However, these approximations are associated with noise induced upon the real problem. The aim is to keep to a minimun this added noise, as this will increase the prediction quality.

We have presented here a mathematical-statistical approach to analyze the spatio-temporal dynamics of brain tumors. They come in form of processed computer tomography images. We interpret them as collections of image pixels with varying degrees of color intensity levels. As such, they can be considered as a stochastic process, and we make use of spatio-temporal stochastic processes as the right statistical framework. Using this framework, we are able to predict cancer growth in space and time, and show real data analysis. The results are shown to be satisfactory, as noted in the prediction shown in Figure 2.12.

In addition, we have implemented a Matlab software. The code is available upon contacting the authors.

We should also note that we have assumed a constant growth, and in some cases, this growth can not be assumed constant, and we should adapt our modeling strategy to the case of acceleration motion. This is clearly subject of a further research.

Chapter 3

Geometric prediction methods of tumor growth[1]

In present day societies, cancer is a widely spread disease that affects a large proportion of the human population, many research teams are developing algorithms to help medics to understand this disease. In particular, tumor growth has been studied from different viewpoints and different mathematical models have been proposed. Our aim is to make predictions about shape growth, where shapes are given as domains bounded by a closed curve in $\mathbb{R}^2$.

These predictions are based on geometric properties of plane curves and vectors. We propose two methods of prediction and a comparison between them is shared. Both methods can be used to study the evolution in time of any 2D and 3D geometrical forms such as cancer skin and other types of cancer boundary. The first method is based on observations in the normal direction to the plane curve (boundary) at each point (normal method). The second method is based on observations at the growing boundaries in radial directions from the "center" of the shape (radius method). The real data consist of at least two input curves that bind a plane domain.

---

[1] This chapter is based on the published paper: "*Two handy geometric prediction methods of cancer growth*" by Vlad et al. (2015)[2]

The evolution in time of some objects is the subject of study of many researchers worldwide. Special attention has been given to cancer, and a way to understand this disease is to know how it evolves over time.

We can find, in reading, many studies about mathematical modeling of tumors, which attempt to predict the growth of tumors from a mathematical point of view.

Williams and Bjerknes (1972)[64] introduced a stochastic model for the spread of cancer cells. Cells, both healthy and diseased, are situated on a planar lattice. Tumor extension is through cell division, one daughter keeps his position, while the other usurps the position of a neighbor; abnormal cells reproduce at a faster rate than normal cells.

One model that has been used to describe tumor growth is the exponential growth model (see Yorke et al. (1993)[69]) and another model used to describe tumor dynamics is a Gompertz curve or Gompertz function (Qi et al. (1993)[57]). This is a type of mathematical model for a time series, where growth is slowest at the end of a time period (Yorke et al. (1993)[69]). A proposed mathematical model based on energy conservation (Universal Law model) was derived to model tumor growth (West et al. (2001)[70]). This model was tested against empirical data and the results fit a variety of in vitro and in vivo data (Guiot et al. (2003)[71]).

The development of tumor models is important as they offer a way to better understand the kinetic growth of malignant tumors which may lead to the development of successful treatment strategies.

Our aim is to propose two simple prediction methods of the shape growth, where shapes are given as bounded domains by a closed curve in $\mathbb{R}^2$.

Let us assume that the cells, both normal and abnormal, are situated on a planar lattice $\mathbb{R}^2$ and let D_t ($D_t \in S$, where $S =$ sample space or Region of Interest) denote a bounded domain occupied by cancer cells at time t. Given the domain $D_t \in \mathbb{R}^2$ we consider that it is bounded by a simple closed curve parameterized by arc length $\alpha_t : [0, L] \longrightarrow \mathbb{R}^2$. So, the original cancerous population occupies the planar domain D_0. For simplicity, Williams and Bjerknes (1972)[64] restricted their attention to the process α_0 starting with a single abnormal cell at the origin. In this context, the following question arises: how fast does α_0 grow, and what is the geometric nature of α_t for time $t > 0$ (but not too large).

To make the prediction we use two geometrical methods and start from the hypothesis that the speed of variation in time (growing) is constant in each direction (but not equal). So the evolution in time of the tumor can be expressed by the variation of each vector. Starting from this, the problem is resumed to determine the values of each vector for a future time $(t + \Delta t)$.

In the first method (normal method) we construct the vectors in the normal direction to the curve and in the second one (radial method) we construct the vectors as an extension of the radius of a circle in which the tumor can be enrolled.

The usual data consist of in at least two curves that bound two growing planar domains. The first curve can be the contour of one tumor when it was discovered (at time t), and the second curve is the same tumor after a while Δt. For real data analysis both curves can be provided by two analysis with computed tomography (CT), at a time interval.

Although these curves are continuous (parameterized) curves, all our calculations, which are based on the comparison of these curves, are done on digital curves, that is, a discretized version of the parameterized curves using contour points of each curve.

The values of the vectors from a specific time can be calculated if we know the parametric function or the discretized step. From both methods of calculus the input data represent the coordinates of the contour points.

In next section we present a brief theory of growth shapes and curve deformations, and we propose our two prediction methods of tumor growing, depending on the direction of growing chosen from each point of the curves.

In Section 3.2 we implement all the mathematical calculations in Matlab software, and we build a library of functions to run both methods. We present the results of stimulation study based on random and parametric curves. We also present the analysis of a real data set.

3.1 Methodology

3.1.1 Shape and growth description

At present, there exists a variety of growth models for objects in discrete space; see for instance Bramson and Griffeath (1981)[55], Qi et al. (1993)[57], Lee and Cowan (1994)[58], and Kansal et al. (2000)[59]. In Richardson (1973)[63], the growth model is described by a Markov process. For a growing object in the plane, the state at time t is a random subset Y_t of $\mathbb{Z}^2$ consisting of the "infected sites", and Y_0 (initial tumor) consists of a single site.

So, it is deduced from the preceding results that the tumor shape Y_t at present time t, depends on the structure of the initial tumor shape Y_0. Then the tumor shape in a future $Y_{t+\Delta t}$, is a function which depends on the edge and structure of the cancer in the present time Y_t, and also on some external factors like mitosis, nature of cancer (benign or malign), density, etc $\big($all these factors can be included in a function $g(t)\big)$ then,

$$Y_{t+\Delta t} = f(Y_t) + g(t) \tag{3.1}$$

In our study we consider the boundary of the tumor at different times (not the entire tissue). This boundary is represented by a closed curve which may be star-shaped or of any other random form. In real studies, a tumor is discovered time after starting its development and, for this reason, the initial domain D_0 does not consist in just one cell but in a closed domain bounded by a curve α_0. So, each domain $D_t \in \mathbb{R}^2$ is bounded by a closed curve α_t and we propose two simple methods to predict the tumor growth.

To predict the tumor growth means to find the curve $\alpha_{t+\Delta t}$ based on observations at times $t_i < t + \Delta t$. Both methods are based on geometric properties of curves and vectors and the main differences between them consist in the directions of the vectors chosen at points $P_i \in \alpha_t(s_i)$.

The first method (normal method) can be applied to general curves (not necessarily star-shaped) but the observed curves must be close enough to avoid self-intersections between normal lines. The second method (radial method) can be applied only to star-shaped domains with respect to a point in D_0.

3.1.2 Normal method

The first method is based on observations in the normal direction to the plane curve (boundary) at each point. We consider negatively oriented planar closed curves $\alpha : I \longrightarrow \mathbb{R}^2$ (that is, when traveling on α one always has the curve interior to the right). Moreover, when the curve α is parameterized by arc length s (see Carmo (1976)[72]); then, the signed curvature function of $\alpha(s)$ is defined as

$$\frac{dT}{ds} = \kappa(s)\, N(s), \tag{3.2}$$

where $T(s)$ is the unit tangent vector and $N(s)$ is the unit normal vector oriented to the exterior of D (like the sunlight).

The sign of the signed curvature κ indicates the direction in which the unit tangent vector rotates as a function of the parameter along the curve. If the unit tangent rotates counterclockwise, then $\kappa > 0$. If it rotates clockwise, then $\kappa < 0$ (see Figure 3.1).

For a plane curve given by a parametrization $\alpha(s) = \big(x(s), y(s)\big)$, the signed curvature is expressed as

$$\kappa(s) = \frac{x'y'' - y'x''}{\big((x')^2 + (y')^2\big)^{3/2}}. \tag{3.3}$$

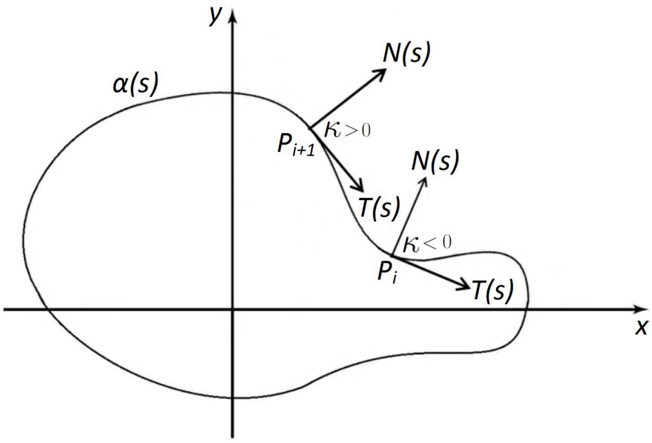


FIGURE 3.1: Signed curvature κ for a negatively oriented planar closed curve α

To describe growth shape, our data consists of a discrete number of shapes $D_0, D_1, \ldots D_n$ obtained at times $t_i = t_0 + i\Delta t$, respectively, and bounded by closed simple curves $\alpha_i, i = 1, 2, \ldots n$, where each curve α_i is defined from the preceding one as

$$\alpha_i(s) = \alpha_{i-1}(s) + f_{i-1}(s)N_{i-1}(s), \quad i = 1, 2, \ldots n-1, \tag{3.4}$$

where $N_{i-1}(s)$ is the unit normal to $\alpha_{i-1}(s)$ and $f_{i-1}(s)$ is a differentiable function.

In order to make a prediction, that is, in order to construct a curve $\alpha_{n+1}(s)$ from the information provided by $\alpha_0, \alpha_1, \ldots, \alpha_n$ and therefore a function $f_n(s)$ from $f_{i-1}(s)$, $i = 1, 2, \ldots n$, we consider the curve:

$$\alpha_{n+1}(s) = \alpha_n(s) + f_n(s)N_n(s), \tag{3.5}$$

where $\alpha_n(s)$ and $N_n(s)$ are defined in (3.4) and we suppose that $f_n(s) = f_{n-1}(s)$, $\forall s \in I$.

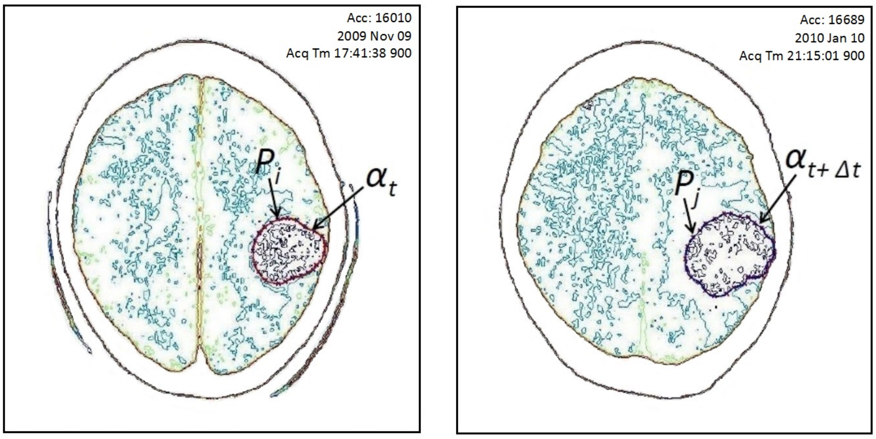
Then, we suppose that $f_i(s) = f_{i-1}(s)$, $\forall s \in I$ and for $i = 1, 2, \ldots n$ which means that the function $f_i(s)$ at time t_i is $f_i(s) = i f_0(s)$.

Note that if $f_i(s) = k$, constant, then $\alpha_{i+1}(s)$ is the parallel curve to $\alpha_i(s)$ at a distance k (see Gray (2004)[73]). This is the simplest case when the evolution in time in each direction is constant and equal to k. Note that to ensure that the curve α_{n+1} is well defined it is necessary that, for all $s \in I$, the distance from the curve α_n to the point $\alpha_{n+1}(s)$ will be $f_n(s)$.

Therefore, in our prediction, we take into consideration that the growing rate is different for each normal direction but constant for equal periods of time. In real studies the tumor is discovered after some time t and instead of a parameterized curve the study is based on digital curves.

In Figure 3.2a we show the digitization of a brain tumor at time t_1 which corresponds to curve $\alpha_1(s)$ plotted at points $P_i^{t_1}(x_i, y_i)$, with coordinates x_i and y_i, $i = 1, \ldots, n$.

After a time Δt, for $t_2 = t_1 + \Delta t$, the same growing tumor has a different contour (Figure 3.2b), $\alpha_2(s)$, represented at points $P_j^{t_2}(x_j)$, $j = 1, \ldots, m$.



a.) b.)

Figure 3.2: Evolution in time of a tumor brain cancer: a) first image acquisition of curve α_1 and a points $P_i^{t_1}$; b) the same tumor after time Δt

The number of points n and m are factors which represent the resolution or the step of digitization for each brain tumor contour.

Note that since the curves are digital we only know a discrete number of points $P_i^{t_1}$, $i = 1, 2, \ldots n$ and $P_j^{t_2}$, $j = 1, 2, \ldots m$ for each curve: $\alpha_1(s_i)$ and $\alpha_2(s_j)$. Then, in general, it is possible that the estimated point $\overline{P_j^{t_2}} \in \alpha_2(s_j)$ with $j \in \{1, \ldots, m\}$ does not correspond to a point $P_j \in \alpha_2(s_j)$. For this reason, we join the points $\overline{P_j^{t_2}(x_j, y_j)}$ by straight segments and we obtain an approximated polygonal curve to $\overline{\alpha_2(s_j)}$ that, as there is no confusion, we will denote it also by $\overline{\alpha_2(s)}$.

Now, from the n points $P_i^{t_1}(x_i, y_i)$ in curve α_1 and an approximation of the values of $N(s_i)$, we will compute the points in the polygonal curve α_2. We will suppose that we have two curves α_1 and α_2 and we will predict, under our assumptions, the curve α_3.

We proceed as follow. We consider three consecutive points $P_{i-1}^{t_1}$, $P_i^{t_1}$ and $P_{i+1}^{t_1}$ and the segment $\overline{P_{i-1}^{t_1} P_{i+1}^{t_1}}$; then we draw the perpendicular line to this segment which passes through the point $P_i^{t_1}$. The intersection between this line and the polygonal curve α_2 corresponds to the calculated point $\overline{P_j^{t_2}}$ and the distance between $P_i^{t_1}$ and $\overline{P_j^{t_2}}$ is $f_1(P_i^{t_1})$. We repeat this process for $i = 1, 2, \ldots, n$ and we obtain all the points of $\overline{P_j^{t_2}}$, for $i = 1, 2, \ldots, n$.

Other methods to approximate the normal direction to a point $P_i^{t_1}$, like the bisector direction or the median direction, can be found in Belyaev et al. (1999)[74].

At each point $\overline{P_j^{t_2}}$ we determine the normal vector $N_2(\overline{P_j^{t_2}})$ to the polygonal curve α_2 and, following that direction, at a distance $f_1(P_i^{t_1})$, we plot the predicted points $\overline{P^{t_3}(x_j, y_j)}$, $j = 1, 2, \ldots, m$, of the predicted curve $\alpha_3(s)$, at time $t_0 + 2\Delta t$ (see Figure 3.3).

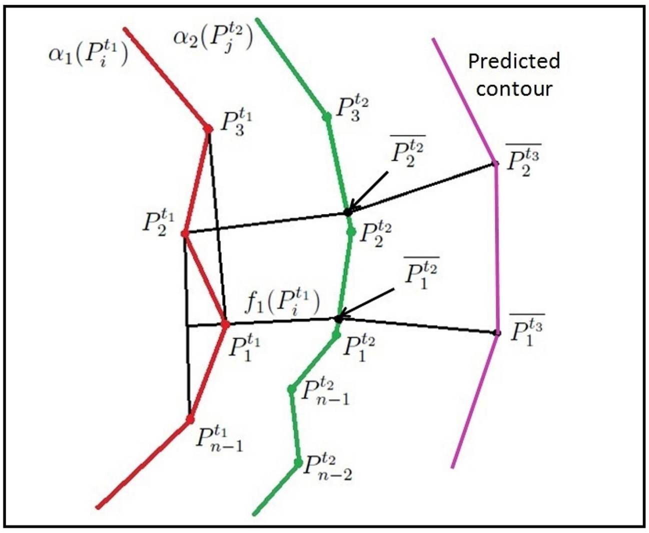


FIGURE 3.3: Calculus of the predicted points $\overline{P_k^{t_3}}$ with the normal method

The condition that must be required to apply this method is that vectors $\overrightarrow{P_i^{t_2} P_j^{t_3}}$ do not intersect between them for $i = 1, 2, \ldots, n$. This can be accomplished when the distance $f_1(P_i^{t_1})$ for each point $P_i^{t_1}$ does not exceed the value of the $\kappa(P_i^{t_1})$ (Gray (2004)[73]). There exist several different points approximation methods for the discrete approximation of the curvature; for instance, circle approximation or angle approximation (see for instance Nitzberg et al. (1993)[75]).

However, we will apply the Archimede's theorem of the area of a parabolic segment to approximate the curvature from a parabola (Gual-Arnau and Monterde (2015)[76]).

Given $P_i^{t_1}$ and the segment $\overline{P_{i-1}^{t_1} P_{i+1}^{t_1}}$ whose length is c (Gual-Arnau and Monterde (2015)[76]), we have that

$$A(\triangle_i) \approx \frac{\kappa_1(P_i^{t_1}) c^3}{12}, \tag{3.6}$$

where $A(\triangle_i)$ is the area of the triangle of vertices $P_{i-1}^{t_1}$, $P_i^{t_1}$ and $P_{i+1}^{t_1}$.

Since the distance d from the point $P_i^{t_1}$ to the line defined by $\overline{P_{i-1}^{t_1} P_{i+1}^{t_1}}$ (see the triangle $\triangle_2 = \triangle P_1^{t_1} P_2^{t_1} P_3^{t_1}$ in Figure 3.3) is:

$$d = \sqrt{a^2 - \frac{a^2 - b^2 + c^2}{2 * c^2}} \tag{3.7}$$

where:

$$\begin{aligned} a &= \| \overline{P_i P_{i+1}} \| = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \\ b &= \| \overline{P_{i-1} P_i} \| = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \\ c &= \| \overline{P_{i+1} P_{i-1}} \| = \sqrt{(x_{i-1} - x_{i+1})^2 + (y_{i-11} - y_{i+1})^2} \end{aligned}$$

The approximation of the curvature when we know the coordinates of the points $P_{i-1}^{t_1}$, $P_i^{t_1}$ and $P_{i+1}^{t_1}$ is, from (3.6),

$$\kappa(P_i^{t_1}) \approx \frac{6\,d}{c^2}. \tag{3.8}$$

Moreover, if we have the approximation of $\kappa_1(P_i^{t_1})$ via (3.8) and that of $f_1(P_i^{t_1})$ we may know if there exists a relation between local curvature and the growth shape. A discussion in this line that has been considered in interesting applications (Brú et al. (1998)[77]).

3.1.2.1 Curve evolutions

This method of predicting growth is a particular case of curve evolution, where each point of a curve α moves in the normal direction with speed equal to the function $f(s)$ at that point. Consider a family of smooth closed curves $\alpha(s,t)$ where t means time and s is the parameter of the curve (s and t are independent) and suppose that $\alpha(s,t_i)=\alpha_i(s)$, for $i=1,\dots,n$.

The mathematical formulation in this case is

$$\frac{\partial\alpha(s,t)}{\partial t}=F(s)N(s,t),\quad i=1,\dots n-1, \tag{3.9}$$

where $N(s,t)$ is the normal vector to the curve $\alpha(s,t)$ and $F(s,t)$ is the speed function. In principle the function F may depend on many factors like local and global properties of the growing curve and time t. However, in our method F does not depend on t.

For numerical implementation we approximate

$$\frac{\partial\alpha_i(s,t)}{\partial t}|_{t_i}\approx\frac{\alpha_i(s,t_i+\Delta t)-\alpha_i(s,t_i)}{\Delta t}=f(s)N_i(s),\quad i=1,\dots n-1. \tag{3.10}$$

Then, since Δt is constat for each i, we have:

$$\alpha_{i+1}(s)\approx\alpha_i(s)+\Delta tf(s)N_i(s)=\alpha_i(s)+f_0(s)N_i(t). \tag{3.11}$$

Several examples of functions $F(s,t)$ can be found in Belyaev et al. (1999)[74] for instance, when the function is the curvature of the initial curve, we have a curvature-driven evolution of the initial curve α_0.

3.1.3 Radius method

In this case we consider star-shaped domains from an origin $O(x_0, y_0)$ and vectors which are represented by the radius line from this origin to the each point $P_i^{t_1}$ of first curve α_1 in that direction.

This method is quite simple to understand from the theoretical viewpoint and also from the point of view of calculus. Starting from the center point of tumor $O = O(x_0, y_0)$ we construct a line to each contour point $P_i^{t_1}$ of the curve α_1 and continue to the intersection of the second contour α_2 which give us the points of the second contour $\overline{P_j^{t_2}}$, for each $i = 1, 2, \ldots, m$ (see Figure 3.4a).

These lines with the direction from the center point O to the points $P_i^{t_1}$ will be called radius vectors and denoted by $\overrightarrow{R_i^{t_1}}$.

The estimated points $\overline{P_j^{t_2}}$ of the second curve $\alpha_{t_1+\Delta t}$, as it is a second time, are: $\overline{P_1^{t_2}}$, $\overline{P_2^{t_2}}$, $\overline{P_j^{t_2}}$, here $j = 1, 2, \ldots, n$. The third curve (the simulated curve) is $\alpha_{t_1+2\Delta t}$ because the time intervals between t_1, t_2 and t_3 are supposed to be equal. These three curves are denoted by α_1, α_2 and α_3 respectively.

For each $j = 1, 2, \ldots, n$ the distances between $P_i^{t_1}$ and $\overline{P_j^{t_2}}$ are the values of the function $f_1(P_i^{t_1})$ and we can calculate the spatial coordinate of the contour points $\overline{P_k^{t_i}}$ with $j = 1, 2, \ldots, n$ at time $t_1 + 2\Delta t$.

3.2 Simulations and application

3.2.1 Simulated data: random curves

The simulation consist in creating three discrete closed curves with different areas, keeping the same center. Curves are created by joining a predetermined number of points arranged in a circle of radius defined by the user plus a uniformly distributed random variable.

Each curve is simulated with a different number of contour points. After generating the curves we apply both prediction methods, and calculate the predicted contour for each method.

Radius method: we start to construct each vector $\overrightarrow{R_i(P_i)}$ from the center $O(x_0, y_0)$ of the first tumor α_t to each contour point $P_i(x_i, y_i)$ at time t, and we continue to the intersection of the second contour $\overline{P_j(x_j, y_j)}$ at time $t + \Delta t$ and so on by the predicted point $\overline{P_k(x_k, y_k)}$.

So as not to overload the picture, we plot just the first two iterations and the corresponding points of calculus $P_1^{t_1}$, $\overline{P_1^{t_2}}$, and the predicted point $\overline{P_1^{t_3}}$ (see Figure 3.4a).

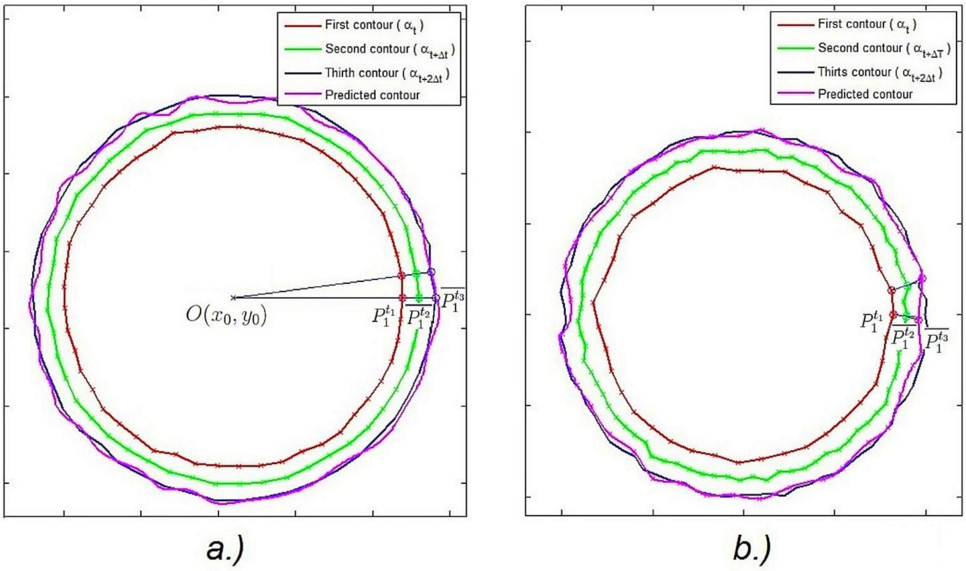


FIGURE 3.4: Simulated tumors at time t, $t + \Delta t$ and $t + 2\Delta t$ using random curves: a) prediction with radius method; b) prediction with normal method

Normal method: in order to begin the simulation, the first vector is the normal vector $\overrightarrow{N_1(P_1^{t_1})}$ to surface α_t in point $P_1(x_1, y_1)$, and in that direction we find the point $\overline{P_j(x_j, y_j)}$ i.e $\overline{P_1^{t_2}}$ at the intersection with the second contour $\alpha_{t+\Delta t}$. Starting from this point in the direction of the normal vector to second contour, we calculate the spatial coordinates of the predicted point (Figure 3.4b).

To show the precision of these methods we repeated simulation 100 times and calculated the values of absolute error (ε_a) and relative error (ε_r) between the area of the contour curve at time $t+2\Delta t$ names as t_3 i.e $\alpha_{t+2\Delta t}(s)$ or $\alpha_3(P_i^{t_3})$ (the area of the polygon determined by the points $P_i^{t_3}$) and the area of the predicted curve (the polygon determined by the estimated points $\overline{P_k^{t_3}}$).

All the calculus proposed in these methods were done with Matlab and did not require too many computing resources. The results are given in Table 3.1.

METHOD	ABSOLUTE ERROR (ε_a)			RELATIVE ERROR (ε_r)		
	min.	average	max.	min.	average	max.
Normal	103.6847	565.9698	1022.6573	0.0033	0.0180	0.0325
Radius	1.6078	195.1009	608.7794	0.00003	0.00368	0.01148

TABLE 3.1: Absolute and relative errors for areas obtained from the normal and radius methods.

In a machine with i3 processor and 4 GB of RAM the time for simulating the growing tumor based on three curves, each one with resolution $n = m = 50$ points, based on normal method was 1.897642 sec, and it is very accurate.

The running time to simulate the growing of the tumor with the radius method, in the same conditions as in the preceding method, was 0.909440 sec, which means a decrease of the computational effort.

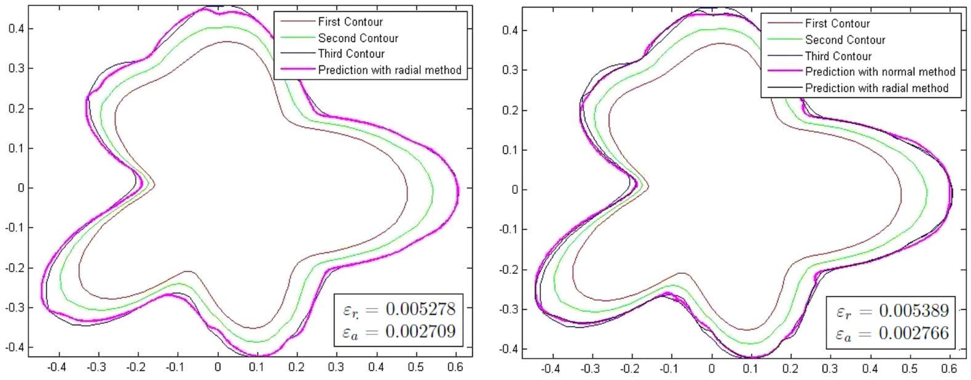
When star-shaped tumors are considered, both methods are accurate in prediction and are computationally fast, and can be used with success in lower computational machines.

Comparing the results we obviously note that the radius method is more effective, and the user requires less recourses from the computing machine, so it is very clear that it is a better method.

3.2.2 Simulated data: parametric curves

To check for the accuracy of each algorithm, we now consider a parametric irregular form and its growth at regular intervals of time. The curves come from a Fourier series expansion of a sine and cosine function.

In Figure 3.5 we plot in red the curve α_{t_1}, in green the curve α_{t_2} and in blue the curve α_{t_3}. Taking into account the first two boundaries of the shape we predict the third curve (drawn in magenta) and we compare with the real evolution of the function.



a.) b.)

FIGURE 3.5: Parametric and simulation shapes of curve at time t_1, t_2 and t_3: a) prediction with radius method; b) prediction with normal method

We also calculated the area of the surface drawn by the predicted contour and the errors absolute and relative reported to the area of surface bounded by curve α_{t_3}:

a) relative error:

radius method: $\varepsilon_r = 0.005278$

normal method: $\varepsilon_r = 0.005389$

b) absolute error:

radius method: $\varepsilon_a = 0.002709$

normal method: $\varepsilon_a = 0.002766$

As can be noted, the absolute and relative errors for the normal method are larger than these for the radius method.

3.2.3 Real data

We now show the performance of the two methods over real data. It was necessary to implement, in one library, a series of functions to process images from a magnetic resonance tomography, computed tomography or any other analysis or image of a tumor. The input data is represented by a complete set of images of a brain tumor taken at intervals of one month, to which we apply the methods to predict its growth.

This particular tumor is found in the Central Nervous System and is called glioblastoma multiform. In conformity with the World Health Organization, this tumor is the most aggressive tumor and the type and grade is according to IV-th classification. In the scans that we are studying it is clear that there is a presence of multiple tumors in the body which is known as metastasis. These metastatic tumors are children of primary tumors from the breast, lung, colon, stomach and skin (melanoma), but in our case, the first was the brain tumor.

From this complete set of analysis we selected three images taken in the same plane: one from November 9^{th}, 2009 (see Figure 3.6a), one from December 8^{th}, 2009 (Figure 3.6b) and the last from January 10^{th}, 2010 (Figure 3.6c).

The input data for this analysis was the first two curves. The boundary tumor of November was digitalized in 64 points: $\alpha_{t_1} = \{P_1^{t_1}, P_2^{t_1}, \dots, P_{64}^{t_1}\}$. The curve is closed, so $P_1^{t_1} \equiv P_{64}^{t_1}$. The second one (corresponding to December) was approximated with 61 points, then: $\alpha_{t_2} = \{P_1^{t_2}, P_2^{t_2}, \dots, P_{61}^{t_2}\}$.

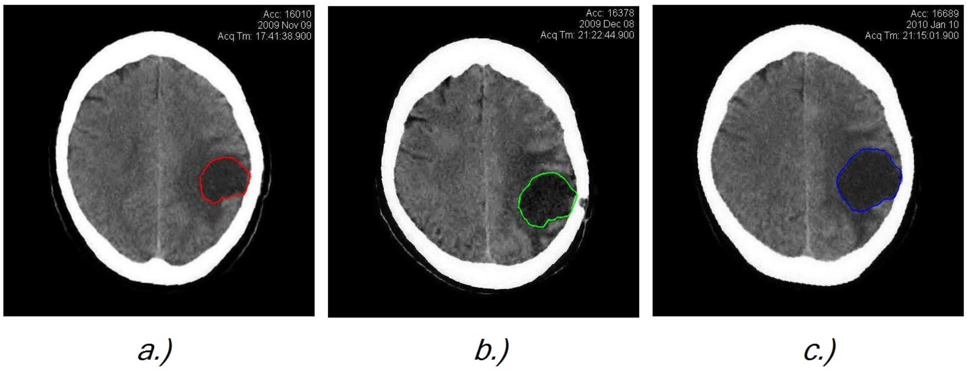


FIGURE 3.6: Evolution in time of the real brain tumor: a) boundary of the tumor at time t; b) boundary of the same tumor in december; c) boundary of the tumor in January

We applied both prediction methods starting from the first two images and we calculated the prediction of growth of this tumor, which we scheduled for January 10^{th}, 2010. We can directly compare the results of the prediction methods, with the third image from the set of analysis.

The reference curve (plotted in blue in Figure 3.7) is the approximation in 60 points of the real contour of brain tumor. Because we start from the first curve and all the calculus are related to the number of points $P_i^{t_1}$, (in this case $i = 64$), the number of points of the prediction is identical, i.e k =64 points.

The area of the surface domain D_3 representing a polygon with 59 sides is $Area(D_3) = 7852.13479$ and the predicted areas are:
a) radius method: $Area(\overline{D_3}) = 7774.37985$;
b) normal method: $Area(\overline{D_3}) = 7776.43387$.

The absolute and relative errors are:
a) radius method:(see Figure 3.7a)
- absolute error: $\varepsilon_a = 77.7549$
- relative error: $\varepsilon_r = 0.00990$

b) normal method:(see Figure 3.7b)
- absolute error: $\varepsilon_a = 75.7009$
- relative error: $\varepsilon_r = 0.00964$

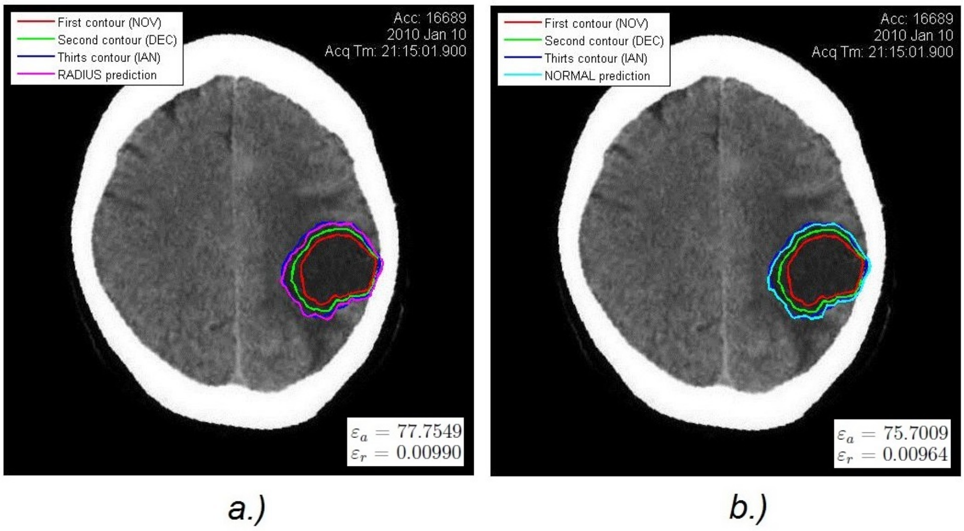


FIGURE 3.7: Brain tumor: real evolution vs. prediction: a) prediction with radius method; b) prediction with normal method

We note that the results are directly affected by the digitization of the curves by the physicians when they choose the contour points of the tumor. Also we must take into account that the resolution of these kinds of images is low (512x512 pixels). We would obtain better results if the input images have quite a better resolution and if we plot the results with a B-spline curve, which gives a better approximation of the predicted curve, defined in the first instance as a polygon.

Chapter 4

Bayesian prediction of tumor growth[1]

In this paper we analyze the spatio-temporal dynamics of brain tumors. These objects are originally processed from computer tomography images, and can be depicted as a collection of image pixels with varying degrees of color intensity levels. We consider spatio-temporal stochastic processes within a Bayesian framework to model spatial heterogeneity, temporal dependence and spatio-temporal interactions amongst the pixels, providing a general modeling framework for such dynamics. We aim at predicting cancer growth in space and time. We analyze real data on brain tumor based on a set of images taken in several time lags.

Knowing the dynamics of a current real system can allow prediction of the values that certain variables can take after a period of time. This can be studied and characterized by using systems of differential equations, their solutions representing the values that can be taken by each variable at particular times.

To solve the complex systems of differential equations, to date, no robust methodology has yet been developed to ensure solutions for

---

[1] This chapter is based on the published paper: "*Bayesian spatio-temporal prediction of cancer dynamics*" by Vlad et al. (2015)[3]

all types of real problems. In this respect, there are alternative methods for approximating solutions at a specific time, in restricted areas. The Bayesian paradigm can be used in this context.

The goal of inference is to make probability statements about unknown quantities using available information. In this sense, the likelihood contains all the relevant information needed for inference. In general we can say that Bayesian inference delivers an integrated approach to perform inference, prediction and decision.

Modern Bayesian models use simulation methods to generate drawings from the posterior distribution. The simulation methods imply an analysis which is performed depending on the data.

Markov Chain Monte Carlo (MCMC) combined with the Stochastic Partial Differential Equation (SPDE) approach, were the motivation for the Integrated Nested Laplace Approximation (INLA) package for the R software.

The library was initiated by Rue and Martino (2007)[78] and subsequently improved through contributions of Rue et al. (2007)[79], Rue et al. (2009)[80], Lindgren et al. (2011)[81], Lindgren and Rue (2013)[82]. At that time the library was used to design spatial models, non-stationary spatial models, spatio-temporal models, and log-Gaussian Cox point process models. These Cox processes play an important role in stochastic geometry as the building blocks of more complicated random set models, and they also serve as instructive simple examples of random sets. This methodology can be successfully applied in epidemiology, environmental risk assessments, ecology, as well as general geostatistics.

We are here interested in modeling the growth of brain tumors. We can find, in reading, a variety of studies about mathematical modeling of tumors. See, for example, Bramson and Griffeath (1981)[55], Cressie (1991)[56], Qi et al. (1993)[57], Lee and Cowan (1994)[58], Kansal et al. (2000)[59], and many others. But in particular our aim is to apply statistical methods to model cancer growth to further obtain prediction maps in space and time. Our method will help the physicians to approximate the growth rate and the regions in which the tumor is growing faster. All these characteristics will help both the physician and the patient to take a decision about his or her treatment, and in the worst case, about his or her future life, in the sense of living in dignity.

The plan of the chapter is the following. Next section presents the data set and its preparation to further modeling strategies. In Section 2 we present the statistical methodology based on a Bayesian framework. We develop several competing models and Section 3 discusses the results.

4.1 Data set

Our input data is formed by three sets of images of a brain tumor, from the same patient, taken with Computer Tomography (CT) at intervals of one month between each of them, as can be depicted in Figure 4.1.

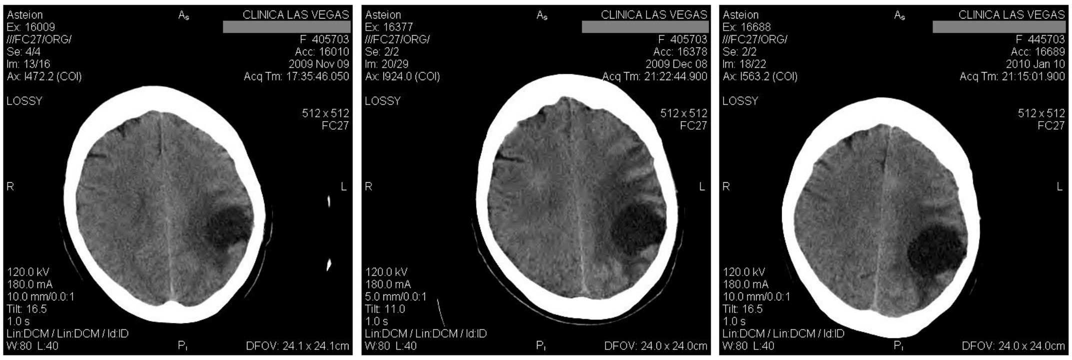


FIGURE 4.1: Original CT image acquisition

This particular tumor is found in the Central Nervous System and is called *glioblastoma multiforme*. According to the World Health Organization, it is the most aggressive kind of tumor and the type and grade belongs to the IV-th classification. In the scans that were studied, it is clear that there were multiple tumors present in the body, called metastasis. These metastatic tumors are the “children” of primary breast, lung, colon, stomach and skin (melanoma) tumors, but in our case, the first was the brain tumor. This is why there are few databases that can be used for this research. The number of CTs taken over time from the same patient with this disease are usually one or two. There are not many cases in which the patient survives to make the third CT. From this complete set of analyses, we selected three images taken in the same plane: one from November 9th 2009, one from December 8th 2009, and the last from January 10th 2010. These images form the input data of our modeling strategy.

The preparation and image cleaning follows two steps. Hence, to use this set of data we must adjust the information taken from the images in order to fit the data. Before image registration, the user needs to know basic information from the image. This basic information consists of obtaining the picture format, the number of colors of the image, the dimension of the matrix, number of pixels, the region of interest and the color intensity for each pixel. With this information at hand, we can thus proceed to calculate basic statistics for each image.

4.1.1 Image registration

We must first locate the cancer tumor and we have to make sure that all three images have the same cartesian coordinates. This is necesary to ensure that the results are based on directly comparable input data which are. To perform all the transformations and calculus required to better locate the cancer tumor and overlap all images, we build a Matlab library presented in the Appendix. The registering and the normalization for all the images that we have for the same tumor is a primary and essential step for further analysis. In these sense we superimpose all the three images using "*overlap.m*" in Matlab. If necessary we can apply some transformations to the images (see description in the List of Matlab functions) as follows:

1. translate the image (use "*transl.m*" function);
2. rotate one or more images to fit the reference image (use "*rot.m*");
3. if one or more images do not overlap the reference after applying these two simple transformations, we can also use an affine transformation of the image (use "*affine_transform.m*" function).

Moreover, for patient identity protection, we cleaned all personal data contained in the CT images. The result of the registration process is shown in Figure 4.2.

Once all three images have a good overlap (the second and third images are fitted to the cartesian coordinates of the reference image), we proceed to normalize the images.

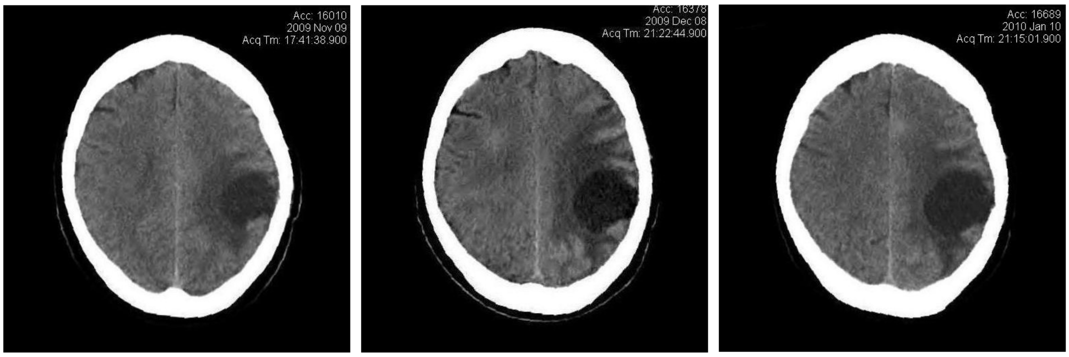


FIGURE 4.2: Registered and normalized images

The normalization process scales the brightness values of the image so that the darkest point becomes black and the brightest point becomes as bright as possible, without altering its hue and contained information. This method can be used for different types of images:

RBG, grayscale or indexed images, with different types of data (uit8, uit16, double). In our case the images are uit8 type and grayscale color. In other words, the normalization process fits the intensity values of the pixels within the range [0, 255]. A value of 0 represents black and 255 refers to white.

This normalization procedure is necessary to ensure that the intensity of color of one particular pixel from the first image is directly comparable with the intensity of color of the same pixel from the second and third images. The intensity of color from each pixel can be an expressed form of cancer cells evolution. For the same reason it is very important that the same pixel has identical cartesian coordinates in all images (the registration procedure helps to ensure that all the images are perfect overlapped).

To do this, we implement a linear interpolation with the "*normaliz.m*" (see description in the List of Matlab functions) Matlab function and we run it for all the three images.

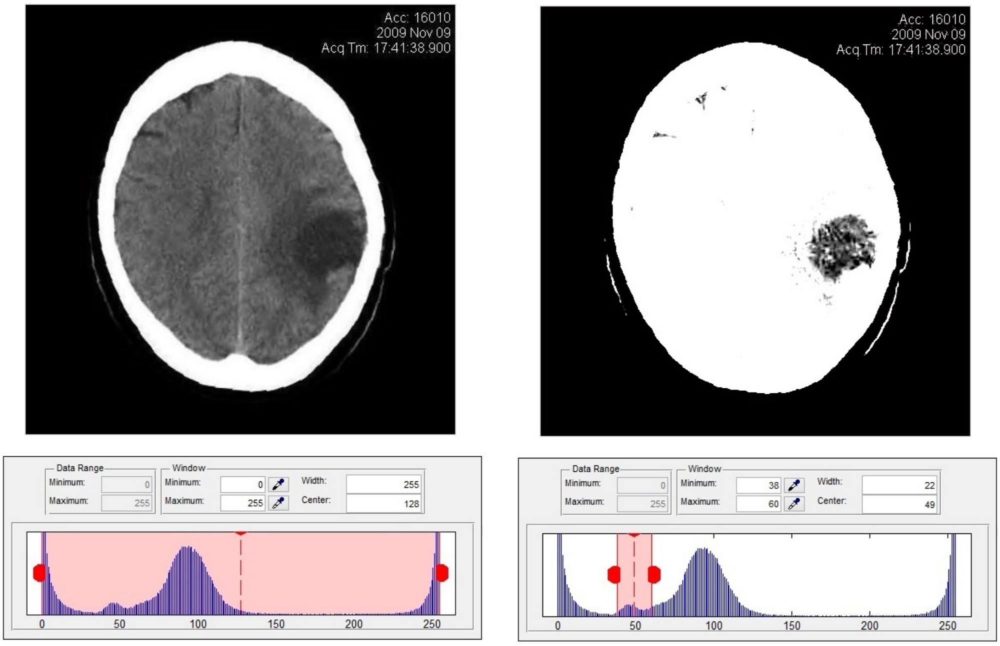
Now having the images registered and normalized, we can move on to the next step: extraction of the needed information.

4.1.2 Preparing the data

In order to better locate the cancer tumor we use the image histogram, also named intensity histogram. With this Statistical Image Tool ([68]), the user will be able to judge the entire intensity distribution at a glance.

The horizontal axis of the graph represents the variations in intensity level, while the vertical axis represents the number of pixels in that particular value. The left side of the horizontal axis represents the black pixels, the middle represents medium gray, and the right-hand side represents light and pure white pixels.

This tool also provides the possibility to modify the plotted range so as to show just the intensity level interval of the desired pixels (thresholding).



Data Range
Minimum: 0
Maximum: 255
Window
Minimum: 0
Maximum: 255
Width: 255
Center: 128

Data Range
Minimum: 0
Maximum: 255
Window
Minimum: 38
Maximum: 60
Width: 22
Center: 49

FIGURE 4.3: Image histogram and thresholding

Because the information contained in the graph is a representation of the pixel distribution as a function of the intensity variation, image histograms can be analyzed for peaks and/or valleys which can later be used to determine a threshold value. This threshold value can then be employed for edge detection, as can be seen in Figure 4.3.

Once located the site of the cancer tumor, we then go on to define the Region of Interest (ROI). The ROI must include the boundary of the cancer tumor. Each point of ROI (i.e., each pixel of the cropped image) can be treated as events of a point pattern with a mark given by the intensity level color, which denotes if that event belongs to the cancerous tissue or not.

The function "*defineROI.m*" (see description in the List of Matlab functions) build in Matlab draws a squared region by defining the length and height, respectively. To avoid overloading the computer by performing unnecessary calculations, we must choose the ROI that is just enough to contain the tumor and the additional tissue under study, but not so small as to not lose the influence of marginal likelihood. In this particular case the input images have the dimension of (512×512) pixels and we define the ROI as a square with opposite points $S_1(280, 410)$ respectively $S_n(172, 302)$ with precisely x-axis, respectively y-axis local coordinates (see Figure 5.4). So, we define the matrix $M_1(x \times y), M_2(x \times y)$ and $M_3(x \times y)$ as ROI with dimensions (130×130).

From the ROI we will extract some information that is needed as input data for further modeling tasks, such as:
1. spatial coordinates of each point from ROI;
2. at least one covariable (e.g., in our case, since we use images as input data the covariable can be the intensity of each pixel);
3. a logical variable zero or one corresponding to the absence or presence of cancer (one if the cell-pixel is cancerous and zero otherwise).

Once the cancer cells have been determined, we can draw the points and the shapes in time of the tumor. The boundaries can be associated with a center of mass and we can compute the distance to each cancer cell.

Before we start the modeling tasks, we proceed to verify the extracted data provided by the images and plot them in the same ROI. As can be seen in Figure 5.4, all infected cells are represented as logical variables and they are well overlapped.

The data set extracted and used for modeling tasks consists of three matrices P_1, P_2 and P_3 (one from each image) with dimension $(n \times m)$, where n represents the number of pixels from ROI, and m are the covariables. The number of pixels are the elements of matrix $M(x \times y)$ of the ROI as can see in Figure 4.4. We note not to make any

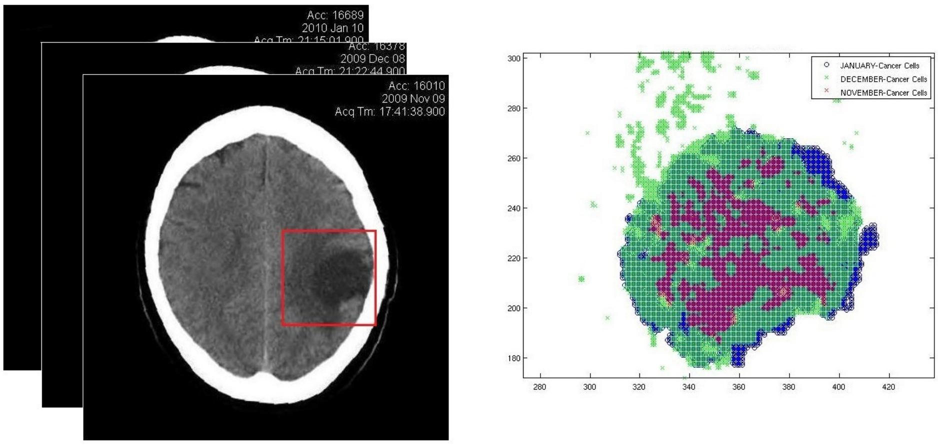


FIGURE 4.4: ROI in registered images (left) and overlapped points for infected cells in ROI (right)

misunderstanding between pixels matrix M_1, M_2 or M_3 with elements $(x \times y)$, when $x = y = 130$ and the matrix P with elements $(n \times m)$, when $n = 16900$ and m is the number of considered covariables. The input data represented by a collection of observations $p = \{p(s_1, t_1), \ldots, p(s_n, t_n)\}$, where the set $(s_1, ..., s_n)$ indicates the spatial locations $(x_1, y_1), \ldots (x_n, y_n)$ at which the measurements are taken, and $(t_1, ..., t_3)$ are the temporal moments.

4.2 Methodology

4.2.1 Statistical framework

Spatio-temporal data can be idealized as realizations of a stochastic process indexed by a spatial and a temporal dimension

$$Y(s,t) \equiv \left\{ y(s,t) | (s,t) \in D \times T \in \mathbb{R}^2 \times \mathbb{R} \right\} \tag{4.1}$$

where D is a (fixed) subset of $\mathbb{R}^2$ and T is a temporal subset of $\mathbb{R}$.

The data can then be represented by a collection of observations $y = \{y(s_1, t_1), \ldots, y(s_n, t_n)\}$, where the set $(s_1, ..., s_n)$ indicates the spatial locations at which the measurements are taken, and $(t_1, ..., t_n)$ are the temporal moments.

The mathematical theory of point processes in a general space is now well established Bremaud (1981)[83], Daley and Vere-Jones (2003)[6]. However, most models for specific applications are restricted either to point processes in time or to two-dimensional space. Cox processes are widely used as models for point patterns which are thought to reflect underlying environmental heterogeneity (Blangiardo et al. (2013)[84] and Cameletti et al. (2013)[85]).

A spatio-temporal correlation structure is a complicated mathematical entity and its practical estimation is very difficult. We thus assume separability in the sense that we model the spatial correlation by the Matérn spatial covariance function and the temporal correlation is modeled using a Random Walk model of order 1 (RW1). We also introduce the interaction effect between space and time using another RW1 structure. Nevertheless, this inclusion of the interaction does not change the separability structure. The Random Walk structure for the temporal dependence is justified by the apparent random distribution over time.

4.2.2 Statistical inference

4.2.2.1 SPDE approach

The SPDE approach allows a Gaussian Field with the Matérn covariance function as a discretely indexed spatial random process which produces significant computational advantages (see for instance Lindgren et al. (2011)[81]). Gaussian Fields are defined directly by their first- and second-order moments and their implementation is highly time-consuming, and gives rise to the so-called "*big n problem*". This is due to the computational costs of $\mathcal{O}(n^3)$ when it comes to performing a matrix algebra operation with $n \times n$ dense covariance matrices, which is notably bigger when the data increase in space and time. To solve this problem, we analyze an approximation that relates a continuously indexed Gaussian Field with Matérn covariance functions to a discretely indexed spatial random process, i.e., a Gaussian Markov Random Field (GMRF).

The idea is to construct a finite representation of a Matérn field by using a linear combination of basis functions defined in a triangulation of a given domain D. This representation gives rise to the stochastic partial differential equation (SPDE), which is a link between the GF and the GMRF. This link allows the spatio-temporal covariance function and the dense covariance matrix of a GF to be replaced with a neighborhood structure and a sparse precision matrix, respectively, both of which are typical elements that define a GMRF. This, in turn, yields substantial computational advantages (Lindgren et al. (2011)[81]).

In particular the SPDE approach consists in defining the continuously indexed Matérn GF, $X(s)$ as a discrete indexed GMRF by means of the representation of a basis function defined on a triangulation of the domain D

$$X\left(s\right) = \sum_{l=1}^{n} \varphi_l(s)\omega_l \tag{4.2}$$

where n is the total number of vertices in the triangulation, $\{\varphi_l(s)\}$ is the set of basis functions, and $\{\omega_l\}$ are zero-mean Gaussian distributed weights. The basis functions are not random, but are instead chosen to be piecewise linear on each triangle

$$\varphi_l\left(s\right) = \begin{cases} 1 & \quad\quad \textit{at vertix } l \\ 0 & \quad\quad \textit{elsewhere} \end{cases} \tag{4.3}$$

The key is to calculate the weights $\{\omega_l\}$, which report on the value of the spatial field at each vertex of the triangle. The values inside the triangle will be determined by linear interpolation (Simpson et al. (2011)[86]).

Thus, the expression (4.2) defines an explicit link between the Gaussian field $X(s)$ and the Gaussian Markov Random Field, and it is defined by the Gaussian weights $\{\omega_l\}$ that can be given by a Markovian structure.

Both the temporal dependence (on t) and the spatio-temporal interaction (on i and t) are assumed to be smoothed functions, in particular Random Walks of order 1 (RW1) (R-INLA project (2012)[87]). Thus, RW1 for the Gaussian vector $x = (x_1, \ldots, x_n)$ is constructed assuming independent increments

$$\triangle x_i = x_i - x_{i-1} \sim N(0, \tau^{-1}) \tag{4.4}$$

The density for x is derived from its $n-1$ increments as

$$\pi\left(x \mid \tau\right) \propto \tau^{(n-1)/2} exp\left\{-\frac{\tau}{2}\sum\left(\triangle x_i\right)^2\right\} = \tau^{(n-1)/2} exp\left\{-\frac{1}{2}x^T Q x\right\}$$

where $Q = \tau R$ and R is the structure matrix reflecting the neighborhood structure of the model (R-INLA project (2012)[87]).

When considering spatio-temporal geostatistical data we need to specify a valid spatio-temporal covariance function defined by

$$Cov\left(y_{it}, y_{jq}\right) = \sigma_C^2 M(s_i, s_j | t, q)$$

where $\sigma_C^2 > 0$ is the variance component and $M(s_i, s_j|t, q)$ is the Matérn spatio-temporal covariance function.

Depending on our assumptions, the spatio-temporal covariance function can be adapted to each situation. In the case of stationarity in space and time, the spatio-temporal covariance function can be specified as a function of the spatial Euclidean distance $\triangle_{ij}$ and of the temporal lag $\Lambda_{tq} = |t - q|$, and so it is defined by

$$Cov\,(y_{it}, y_{jq}) = \sigma_C^2 M(\triangle_{ij}; \Lambda_{tq})$$

If we assume separability, the spatio-temporal covariance function is given by

$$Cov\,(y_{it}, y_{jq}) = \sigma_C^2 M_1(\triangle_{ij}) M_2(\Lambda_{tq})$$

with M_1 and M_2 being the spatial and temporal correlation functions, respectively. Alternatively it is possible to consider a purely spatial covariance function given by

$$Cov\,(y_{it}, y_{jq}) = \sigma_C^2 M(\triangle_{ij})$$

when $t = q$ and 0 otherwise. In this last case, the temporal evolution could be introduced assuming that the spatial process evolves in time following autoregressive dynamics (Harvill (2010)[88]).

Assuming separability, we need to define the Matérn spatial covariance function which controls the spatial correlation at distance $\|h\| = \|s_i - s_j\|$, and this covariance is given by

$$M\,(h \mid \nu, \kappa) = \frac{2^{1-\nu}}{\Gamma\,(\nu)} (\kappa \,\|h\|)^{\nu} \mathrm{K}_{\nu}(\kappa \,\|h\|) \qquad (4.5)$$

where K_{ν} is a modified Bessel function of the second kind and $\kappa > 0$ is a spatial scale parameter whose inverse, $1/\kappa$, is sometimes referred to as a correlation length. The smoothness parameter $\nu > 0$ defines the Hausdorff dimension and the differentiability of the sample paths (Gneiting et al. (2010)[89]). Specifically, we tried ν=1,2,3.

Using the expression defined in (4.5), when $\nu + d/2$ is an integer, a computationally efficient piecewise linear representation can be constructed by using a different representation of the Matérn field $x(s)$, namely as the stationary solution to the SPDE (Simpson et al. (2011)[86]).

$$\left(\kappa^2 - \triangle\right)^{\alpha/2} x(s) = W(s) \tag{4.6}$$

where $\alpha = \nu + d/2$ is an integer, $\triangle = \sum_{i=1}^{d} \frac{\partial^2}{\partial s_i^2}$ is the Laplacian operator, and $W(s)$ is spatial white noise.

In the general spatial point process context, the intensity stands for the number of events (infected cells in our case) per unit area. When considering the total intensity in each cell, we refer to the number of infected pixels per cell area. A particular problem in our dataset is that the total intensity in each cell, Λ_{it} is difficult to compute, and so instead we use the approximation, $\Lambda_{it} \approx |s_i| \exp(\eta_{it}(s_i))$, where $\eta_{it}(s_i)$ is a "representative value" (i.e., it represents the intensity or number of infected pixels in a particular cell given by a linear predictor of covariates and other terms) (Simpson et al. (2011)[86]), within the cell and $|s_i|$ is the area of the cell s_i.

To treat these kinds of problems, Cox processes are widely used. In particular, Log-Gaussian Cox processes (LGCP), which define a class of flexible models, are particularly useful in the context of modeling aggregation relative to some underlying unobserved environmental field (Illian and Hendrichsen (2010)[90]; Simpson et al. (2011)[86]) and they are characterized by their intensity surface being modeled as

$$\log\left(\lambda(s)\right) = x(s) \tag{4.7}$$

where $x(s)$ is a Gaussian random field as in (4.2).

4.2.2.2 Bayesian computation

In a statistical analysis, to estimate a general model it is useful to model the mean for the i-th unit using an additive linear predictor, defined on a suitable scale

$$\eta_i = \beta_0 + \sum_{m=1}^{M} \beta_m z_{mi} + \sum_{l=1}^{L} f_l(v_{li}) \tag{4.8}$$

where β_0 is a scalar which represents the intercept, $\beta = (\beta_1, .., \beta_M)$ are the coefficients which quantify the effect of some covariates $z = (z_1, .., z_M)$ on the response, and $f = \{f_1(.), .., f_L(.)\}$ is a collection of functions defined in terms of a set of covariates $v = (v_1, .., v_L)$.

From this definition, by varying the form of the functions $f_l(.)$ we can estimate different kind of models, from standard and hierarchical regression, to spatial and spatio-temporal models (Rue et al. (2009)[80]). Given the specification in (4.8), the vector of parameters is represented by $\theta = \{\beta_0, \ \beta, \ f\}$.

Our response variable is the pixel intensity, and based on Figure 4.3 we consider that an intensity between 38 and 60 was indicative of an infected pixel. So considering s_i the pixel, and $\eta_{it}(s_i)$ the value of its intensity at time t, we specify the log-intensity of the Poisson process by a linear predictor (Illian et al. (2012)[91]). The most simple model ($M1$) is such that at each time $t = \{Nov, Dec, Jan\}$, we have

$$\eta_{it}(s_i) = \beta_0 + S_i \tag{4.9}$$

where β_0 represents the heterogeneity accounting for variation in relative infected risk across different brain regions and S_i is the spatial dependence. Starting from this first simple model (4.9) we can define more complex models if we add covariables.

In $M2$ with times $t = \{Nov, Dec, Jan\}$ we add the distance X_i from the center of mass of the cancer tumor to each infected pixel

$$\eta_{it}(s_i) = \beta_0 + \beta_1 X_i \tag{4.10}$$

and where β_1 represents the coefficient that quantifies the effect of the distance in the response.

The third model $M3$ has both spatial S_i and temporal τ_t components

$$\eta_{it}(s_i) = \beta_0 + S_i + \tau_t \tag{4.11}$$

Model $M4$ represents the previous model defined by (4.11), but adding a distance covariable

$$\eta_{it}(s_i) = \beta_0 + \beta_1 X_{it} + S_i + \tau_t \tag{4.12}$$

Now the more complex model $M5$ includes all the previous models as well as the interaction between space and time expressed in the following equation

$$\eta_{it}(s_i) = \beta_0 + \beta_1 X_{it} + S_i + \tau_t + \upsilon_{it} \tag{4.13}$$

where β_0 represents the heterogeneity accounting for the variation in relative risk across different infected cells, X_i is the distance from the center of mass to each infected pixel, S_i is the spatial dependence, τ_t is the temporal dependence, and υ_{it} is the spatio-temporal interaction. Note that we assume separability between the spatial and temporal patterns and allow for interaction between the two components.

Following the Bayesian paradigm, we can obtain the marginal posterior distributions for each of the elements of the parameters vector

$$p\left(\theta_i|y\right) = \int p\left(\psi|y\right) p\left(\theta_i|\psi, y\right) d\psi \tag{4.14}$$

and (possibly) for each element of the hyper-parameters vector

$$p\left(\psi_k|y\right) = \int p\left(\psi|y\right) p d\psi_{-k} \tag{4.15}$$

Thus, we need to compute:
(i) $p\left(\psi|y\right)$, from which all the relevant marginals $p\left(\psi_k|y\right)$ can be obtained,
(ii) $p\left(\theta_i|\psi, y\right)$, which is needed to compute the marginal posterior for the parameters.

The INLA approach exploits the assumptions of the model to produce a numerical approximation to the posteriors of interest, based on the Laplace approximation (Tierney and Kadane (1986[92]).

Operationally, INLA proceeds by first exploring the marginal joint posterior for the hyper-parameters $\hat{p}(\psi|y)$ in order to locate the mode; a grid search is then performed and produces a set G of "relevant" points $\{\psi^*\}$ together with a corresponding set of weights, $\{w_{\psi^*}\}$ to give the approximation to this distribution. Each marginal posterior $\hat{p}(\psi^*|y)$ can be obtained using interpolation based on the computed values and correcting for (probable) skewness, e.g., by using log-splines. For each ψ^*, the conditional posteriors $\hat{p}(\theta_i|\psi^*, y)$ are then evaluated on a grid of selected values for θ_i and the marginal posteriors $\hat{p}(\theta_i|y)$ are obtained by numerical integration (Blangiardo et al. (2013)[84]).

$$\hat{p}(\theta_i|y) \approx \sum_{\psi^* \in G} \hat{p}(\theta_i|\psi^*, y)\hat{p}(\psi^*|y)w_{\psi^*} \tag{4.16}$$

Given the specification in (4.13), the vector of parameters is represented by $\theta_i = \{\beta_0, \beta, S, \tau_t,\ \upsilon_{it}\}$, where we can consider $x_i = (S, \tau_t,\ \upsilon_{it})$ as the i-th realization of the latent GF $x(s)$ with the Matérn spatial covariance function defined in (4.5). We can assume a GMRF prior on θ, with mean 0 and a precision matrix Q. In addition, because of the conditional independence relationship implied by the GMRF, the vector of the hyper-parameters $\psi = (\psi_S, \psi_\tau, \psi_\upsilon)$ will typically have a dimension of order 4 and thus will be much smaller than θ. The heterogeneity was specified as a vector of independent and Gaussian distributed random variables on i, with constant precision (R-INLA project (2012)[87]).

Note that in both parts of the model we control for heterogeneity, spatial dependence, and spatio-temporal extra-variability. Models are estimated using Bayesian inference for GMRF through the INLA.

The use of INLA and the SPDE algorithms yields massive savings in computational time and allows the user to work with relatively complex models in an efficient way. All analyses are carried out using the R freeware statistical package (version 3.1) (R Development Core Team (2011) [93]) and the R-INLA package (R-INLA project (2012)[87]).

We have used the conjugate prior to the Poisson likelihood, which is a Gamma distribution function. Indeed, with the aim of checking the robustness of our methodological choice we have used several other (non-conjugate) priors for the precision parameters (in particular Gaussian and flat priors), and the posterior distribution for the precision hyper-parameters has not changed significantly.

In our models, we have include the Gamma conjugate priors used generally in INLA. The priors are specified on the log of the unstructured effect precision, $log\tau \sim logGamma(1, 0.0005)$ and of the log of the structured effect precision $log\tau \sim logGamma(1, 0.0005)$.

It is important to recall that the precision is defined as $\tau = \frac{1}{\sigma^2}$. (Blangiardo et al. (2013)[84] and Cosandey-Godin et al. (2014)[94]). In Simpson et al. (2014)[95], we can find an extensive study of selection of priors for the models. Different ways for the selections are presented and they develop a new method for use the PC-priors in case of very complicated models. In our case, the models are not that complex within the INLA framework, and thus we can use the default parameters. The presented models could be further developed by choosing the PC-priors.

4.3 Modeling results

As we have a battery of competing models ($M1$ to $M5$, as commented in previous section), we compare them using the Deviance Information Criterion (DIC) (Spiegelhalter et al. (2002)[96]), which is a Bayesian model comparison criterion given by

$$DIC = \text{``}goodness\ of\ fit\text{''} + \text{``}complexity\text{''} = D(\overline{\theta}) + 2p_D \tag{4.17}$$

where $D(\overline{\theta})$ is the deviance evaluated at the posterior mean of the parameters and p_D denotes the *effective number of parameters*, which measures the complexity of the model (Spiegelhalter et al. (2002)[96]).

When the model is true, $D(\overline{\theta})$ should be approximately equal to the *effective degrees of freedom*, $n - p_D$. DIC may underpenalize complex models with many random effects.

	M1_Nov	M1_Dec	M1_Jan	M2_Nov	M2_Dec	M2_Jan	M3	M4	M5
DIC	112386.80	110154.95	109176.50	112393.48	110157.38	109181.96	470726.39	470723.51	368123.89
CPO	3.093191	3.217313	3.123008	3.093127	3.217338	3.12307	4.602935	4.602936	3.341009
nEp	2268.31	3817.56	2256.23	2274.85	3820.65	2259.99	2626.58	2621.20	17579.83

Table 4.1: DIC, CPO and nEp for each model

We first show the summary results related to goodness-of-fit for all five models in Table 4.1. This table shows the conditional predictive ordinate (CPO) (Pettit (1990)[97]; Geisser (1993)[98]; Held et al. (2009)[99]), which expresses the posterior probability of observing the value (or set of values) of y_i when the model is fitted to all data except y_i

$$CPO_i = \pi \left(y_i^{obs} \mid y_{-i} \right) \tag{4.18}$$

Here, y_{-i} denotes the observations y with the i-th component removed. This facilitates computation of the cross-validated log-score (Gneiting and Raftery (2007)[100] for model choice $(-(mean\,(log(cpo))))$.

Therefore, the lowest values of DIC and $(-(mean\,(log(cpo))))$ suggest the best fitted model. The last line in Table 4.1 shows the effective number of parameters (nEp) of the model. The larger this is, the worse the data fits the model. A high number of parameters means more complexity. The best models are those with a high level of complexity and a high goodness-of-fit. In general, we choose that model showing the lower CPO and DIC.

All analyses were carried out using the R freeware statistical package (version 3.1) (R-Development Core Team (2011) [93] and R-INLA (2012)[87]).

Fixed effects for all the five models (and denoted by β_0) are expected to have a systematic and predictable influence on data. Computing standard errors of the fixed effects involves the inversion of a matrix or other computationally demanding calculations to obtain the diagonals of the inverse of a matrix.

When the matrix is large, estimating the standard errors is computationally very demanding. For this reason many procedures do not provide standard error estimates for the fixed effects. In our case, the Bayesian approach based on Laplace approximation provides a good and fast procedure to calculate standard errors for each model.

	M1_Nov	M1_Dec	M1_Jan	M2_Nov	M2_Dec	M2_Jan	M3	M4	M5
Mean	4.4042	4.0205	4.2386	3.5939	2.5476	3.2087	4.102	2.9470	2.7470
St_Dev	0.2663	0.4105	0.3641	0.3668	0.6992	0.4567	5.6533	1.6587	3.8353
0.025quant	3.8616	3.18	3.483	2.8367	1.0176	2.2450	-7.5037	-0.619	-6.3689
0.975quant	4.9465	4.8605	4.9933	4.3007	3.8227	4.0770	14.5683	6.3054	11.2738

TABLE 4.2: Fixed effects: Intercept

The fixed effects given by the intercept β_0 are given in Table 4.2. We note that β_0 is only significative for models $M1$ and $M2$, but not for $M3$, $M4$ and $M5$.

We now add the covariable distance as an additional fixed effect as the distance from pixel to the center of the tumor is of interest. The results are shown in Table 4.3.

	M2_Nov	M2_Dec	M2_Jan	M3	M4	M5
Mean	0.0137	0.0250	0.0175	-	0.0219	0.0249
St_Dev	0.0051	0.0099	0.0061	-	0.0062	0.0065
0.025quant	0.0039	0.0070	0.0059	-	0.010	0.0126
0.975quant	0.0242	0.0465	0.0302	-	0.0344	0.0382

TABLE 4.3: Fixed effects: Distance

Note that, compared to the most simple previous models, without the fixed effect distance, the variation that is associated with the random effect increases considerably. The effect of adding the distance to the explanatory variables as done with models $M4$ and $M5$, is clearly significative in predicting the intensity level and thus the probability of a pixel being infected.

	M1_Nov	M1_Dec	M1_Jan	M2_Nov	M2_Dec	M2_Jan	M3	M4	M5
Correlation Coefficient	0.9078	0.9080	0.9183	0.9079	0.9081	0.9083	0.9431	0.9431	0.9931
RMSE	3.0932	3.2173	3.1230	3.0931	3.2173	3.1231	4.6029	4.6029	3.3410

TABLE 4.4: Correlation coefficients for each model

We calculated the correlation coefficient between the predicted and observed values for all the models as shown in Table 4.4. We also calculated the root mean square error (RMSE).

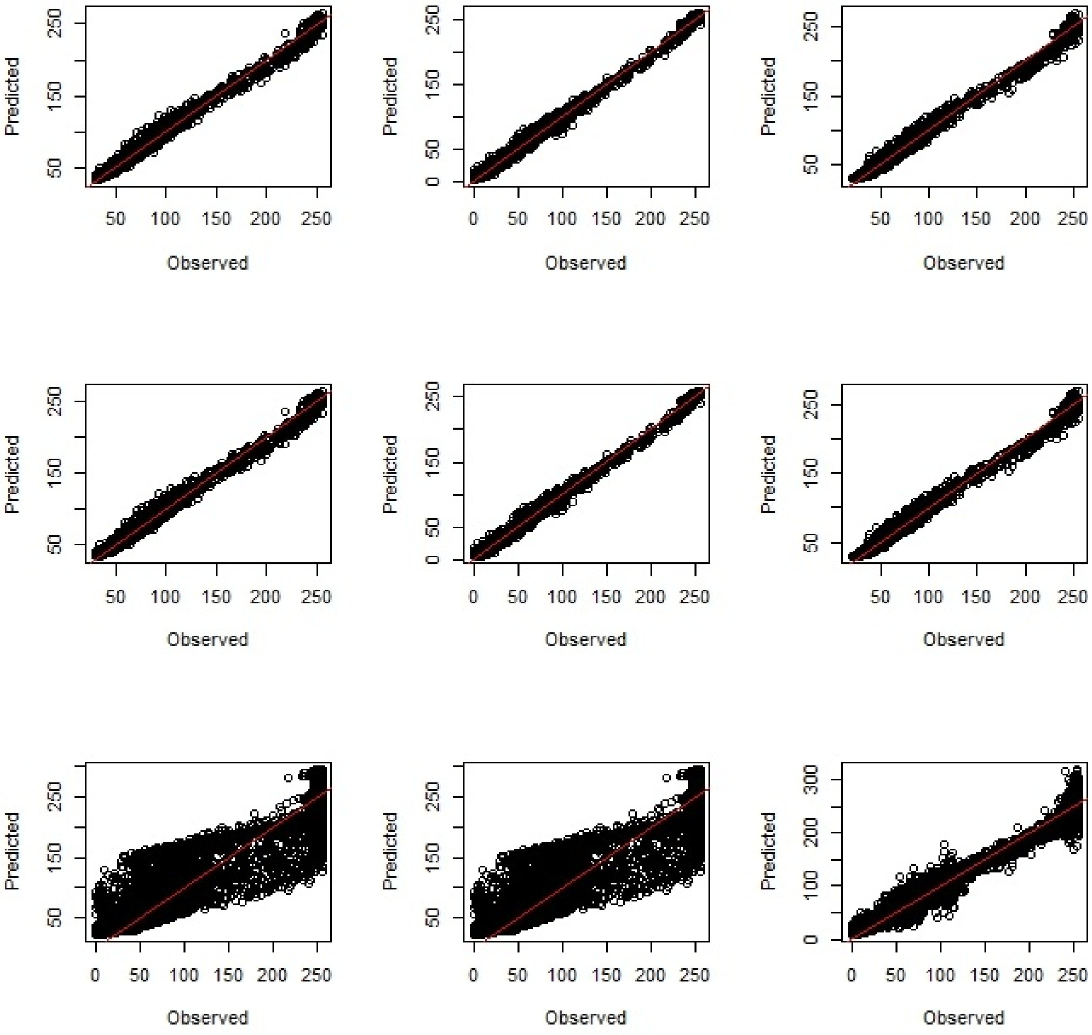


FIGURE 4.5: Correlations between the predicted and observed data for all models. From top left to bottom right: Model $M1$ for November, model $M1$ for December, model $M1$ for January, model $M2$ for November, December, January, model $M3$ (bottom left), model $M4$ (bottom middle) and the complete model $M5$ with covariable distance (bottom right).

Table 4.4 and Figure 4.5 highlight that model $M5$ is the one that best fits the data.

For this model we show in Figure 4.6 the posterior distribution of the parameters playing a role in the model. We note that this model reports spatial interaction, temporal interaction and spatio-temporal one.

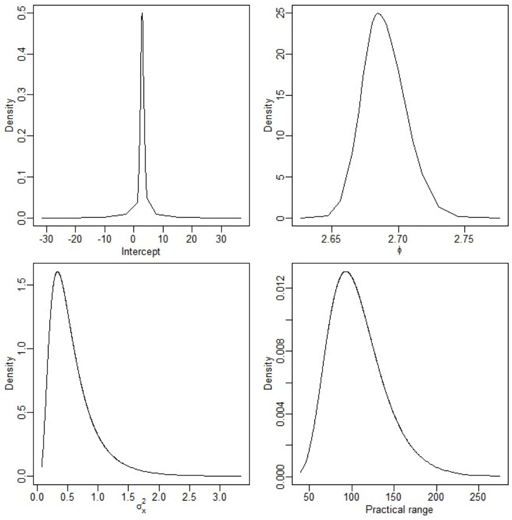


FIGURE 4.6: Posterior distribution of parameters for model M5: $Intercept = 2.7470$ (top left); $\phi = 2.68945162$ (top right); $Normal\ Variance(\sigma^2_X) = 0.5815021$ (bottom left); $Practical\ Range = 107.1021$ (bottom right).

Figure 4.7 shows the prediction map using this model $M5$, highlighting the regions where the cancer has a higher probability of extending in some future time.

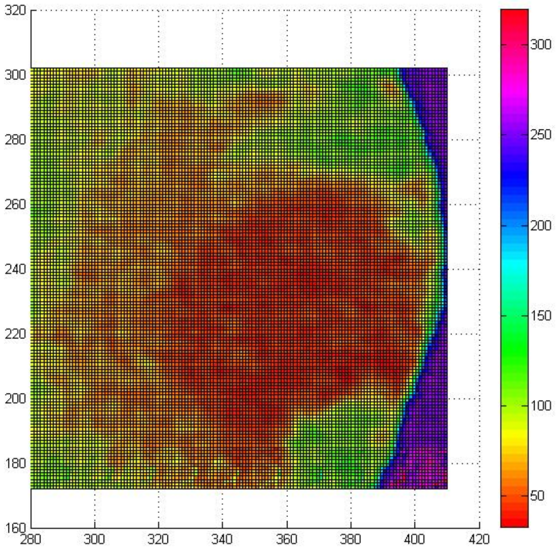


FIGURE 4.7: Prediction map based on model $M5$

Chapter 5

Functional prediction of tumor growth[1]

In this chapter we perform a functional data analysis approach for the investigation of brain tumor contour deformation. The analysis is conducted on a dataset of contour functions extracted from computer tomography (CT) images from patients affected by Gioblastoma multiform (GBM). The procedure follows three steps: first we fit the raw data and we define a global registration criterion for finding an optimal wrapping function for aligning the two set of contours; second we observe and capture the deformation contour function; third step is to recover the shape or pattern of the data from the collected observations.

The plan of the chapter is the following. Next section deals with the problem of registration of contour functions of Gioblastoma Multiform. Section 2 and 3 present our methodological approach. Section 4 presents the application and section 5 describe the results.

---

[1] This chapter is based on the submitted paper: "*Principal differential analysis for modeling dynamic contour evolution. A distance-based approach for the analysis of Gioblastoma Multiform*" by Romano et al. (2014)[4]

5.1 Parametric contour functions of Gioblastoma Multiform: the problem of registration by FDA

Gioblastoma multiform is the most aggressive of the gliomas tumors arising from glia within the central nervous system. Because most patients with this pathology die in less than a year and essentially none has long-term survival, these tumors have drawn significant attention. Several mathematical models for studying the dynamics of the cancer progression have been proposed (see Bauer et al. (2013)[101], for example). These models provide a mathematical expression of the dependence of the tumor size on time. Most of them show that any type of developing tumor has most of its proliferation constrained to the border (Hobolth et al. (2003)[102]). We thus aim at studying what affects the measured contours into different steps of observation. In particular we focus on the problem of monitoring the dynamics of the tumor contour growth.

The tumor contour functions are extracted by a registration algorithm (Vlad et al. (2015)[2]) from CT. The theory related to functions describing contours comes from the shape analysis context (see Kindratenko (2003)[103]). We analyze contour functions by Functional Data Analysis treated as two-dimensional functional data (Epifanio (2011)[104] and Ramsay and Silverman (2005)[105]).

The dataset analysed here is composed by 15 brain tumor contour functions. One of them is based on real data analysis and from the rest we complete the data set with simulated closed curves. We perform an explorative functional analysis of the dataset. We first fit the raw data and then we define a global registration criterion to find an optimal wrapping function for aligning the two set of contours.

The set of data has already been previously analysed with different approaches in two previous exploratory works (Romano et al.

(2014)[106], [107]) with the aim to exploring only the dynamics evolution from the first to the second step.

Functional variability of the two set of contours is assessed by including the derivatives and their relationship in a Principal Differential Model (Ramsay and Silverman (2005)[105]).

Principal differential analysis estimates low-dimensional functional variation as principal component analysis by estimating a differential operator rather than a projection operator. Since we are interested in how the contour functions vary from one replication to another and longitudinally from one observation to another in the two steps of observation, we propose to explore data by a Principal Differential Equation model and to introduce a distance among two principal equation models related to the two stages.

A Principal differential equation model includes estimation of a set of functional coefficients. Then it is applied to the second step of observation. In addition, by fitting a new model in the second step, quantitative information about the degree of local similarity among contour functions and their evolution is found by defining a distance among the coefficients of the two Principal differential models.

We are interested in knowing whether there are any changes in the shape of the tumor contour functions over a real data set. Thus the first main issues consist of recovering the shape or pattern of the data from the collected observations.

Let n be a set of individuals on which we monitor the brain tumor boundary evolution in two different steps of observation. Tumor contours are extracted by an automatic procedure for tumor image segmentation (Romano et al. (2014)[107]). Brain tumor outlines can be seen as a sampled closed contour of a figure in an Euclidean space. Let the perimeter of the figures be S. Every point p_s of the contours can thus be located with coordinates $(X_i(s), Y_i(s)), i = 1, \dots, n$ in a first visit and $(X_i^*(s), Y_i^*(s)), i = 1, \dots, n$ in a second visit. The set of contours can be identified by a set of closed curves, as depicted in Figure 5.1.

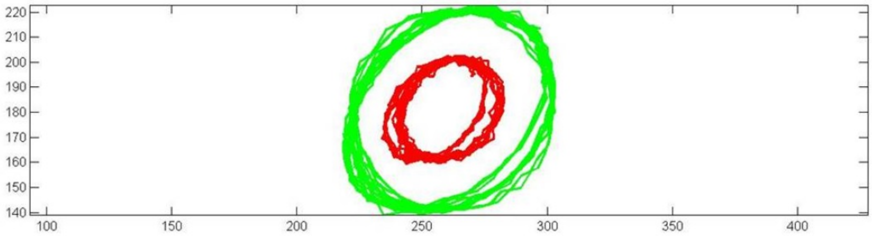


Figure 5.1: Contour function. The red line identifies the first step of observation, the green one the second step

As in shape analysis, several problems arise when comparing tumor contour functions. We consider that sampled functional data $(X_i(s), Y_i(s))$,$i = 1, \dots, n$ and $(X_i^*(s), Y_i^*(s)), i = 1, \dots, n$ can be expressed in terms of K known basis functions. We chose in particular Fourier basis. Thus K couple of vectors of parameters α and β and α^* and β^* are estimated by least squares fitting. The fit of the basis function is not penalized since the differences among the contours depend on the curvature.

Each curve is determined by the coefficients in these basis, and each function is computable for any argument value. This has been performed by means of FDA library (Ramsay and Silverman (2005)[105]). Welch's T test on the mean curvature of the first set of curves $(p - value = 0.0004)$ underlines that the mean curvature in a first stage is significantly smaller than that of the second step.

Then, we need to scale the object to the same surface, to center each profile function and to define a global registration criterion for finding the shift δi for each curve in both the two steps. We scale the shape contour to the same surface S_T. We fix an anticlockwise direction of rotation and standardise the profiles so that we can avoid fictitious variability.

Visual inspection of the first derivative shows that the data presents amplitude variability into the two steps and phase variability among the two steps. We thus look for the optimal wrapping functions maximizing the similarity between the curves in both the two steps and a target curve.

The mean curve of the first set of curves (the “not deformed” curves) is used as target curve for the registration. We define the following global registration criterion in order to search for a shift δi for each curve i with respect to the target curve

$$\Delta = \sum_{i=1}^{N} \int_{S} [(x_i(s+\delta i) - \bar{x}*(s))^2 + (y_i(s+\delta i) - \bar{y}*(s))^2 ds] \quad (5.1)$$

Each curve is then shifted so as to minimize Δ. The estimated means are then updated by re-estimating them from the registered objects.

In order to have the same number of points for all functions, we evaluate the functions in 100 equidistant points from 0 to 1. We therefore have two pairs of functions (representing coordinates) $(X_i(s), Y_i(s))$ and $(X_i^*(s), Y_i^*(s))$ for each individual, with $s \in [0, 1]$.

The derivatives of functional observations play a role for this kind of data as shown in this first explorative step. Now we look how velocity of $(X_i(s), Y_i(s))$, when $i = 1, \ldots, n$ may be employed in describing changes of contour functions in the two steps of observation by Principal Differential Analysis (Ramsay and Silverman (2005)[105]).

5.2 Principal differential analysis and tumor growth

Principal Differential Analysis extends the concept of differential equation into the framework of functional data. In the context of tumor growth we view the contour functions as a dynamic system described by the linear relations among the derivatives. Because the structure of our data is sinusoidal it seems appropriate to set out a second order differential equation model that can capture contour function dynamics.

We define the following non-homogenous differential operator for the couple of functional contours in the two steps (for simplicity we report only the ones related to the first step):

$$\begin{aligned} LX(s) &= \alpha_x(s) + \epsilon_x(s) \\ LY(s) &= \alpha_y(s) + \epsilon_y(s) \end{aligned} \tag{5.2}$$

where $LX(s)$ and $LY(s)$ are defined as

$$\begin{aligned} LX(s) &= \beta_{1x}(s)DX(s) + \beta_{2x}(s)D^2X(s) + D^3X(s) \\ LY(s) &= \beta_{1y}(s)DY(s) + \beta_{2y}(s)D^2Y(s) + D^3Y(s) \end{aligned} \tag{5.3}$$

and where the differential operator D can be written as

$$\begin{aligned} D^3X(s) &= \alpha_x(s) + \beta_{1x}(s)DX(s) + \beta_{2x}(s)D^2X(s) \\ D^3Y(s) &= \alpha_y(s) + \beta_{1y}(s)DY(s) + \beta_{2y}(s)D^2Y(s) \end{aligned} \tag{5.4}$$

The two set of contours modeled by a linear differential operator are respectively characterised by six functions $\alpha_x(s)$,$\alpha_y(s)$,$\beta_{1x}(s)$, $\beta_{2x}(s)$, $\beta_{1y}(s)$, $\beta_{2y}(s)$ for the first step and $\alpha_{x*}(s)$,$\alpha_{y*}(s)$,$\beta_{1x*}(s)$, $\beta_{2x*}(s)$, $\beta_{1y*}(s)$, $\beta_{2y*}(s)$ for the second step. The covariates $\alpha_x(s)$,$\alpha_y(s)$, $\alpha_{x*}(s)$,$\alpha_{y*}(s)$ are the forcing functions and $\beta_{1x}(s)$, $\beta_{2x}(s)$, $\beta_{1y}(s)$, β_{2y},$\beta_{1x*}(s)$, $\beta_{2x*}(s)$, $\beta_{1y*}(s)$, $\beta_{2y*}(s)$ are the weight functions with $s \in S = [0, 1]$.

Each derivative for both the components X and Y has a physical meaning that can help the interpretation of the dynamic changes. $D^X(s)$ can be seen as the velocity contour of the $i - th$ patient; $D^2X(s)$ can be seen as the acceleration contour of the $i - th$ patient; $D^3X(s)$ is the wide margin contour of the $i - th$ patient. We look at linking derivatives and function values together so as to describe by the coefficients the relationships among the physical characteristics.

Thus $\alpha_x(s)$ is a space-varying intercept, $\beta_{1x}(s)$ is the space-varying coefficient relating velocity to the wide margin contour and $\beta_{2x}(s)$ is a space-varying coefficient relating acceleration to wide margin contour. This stands for the X coordinate, but the same applies for the Y coordinate.

A least square criterion is used for estimating these functions, in particular a 34 B-Spline basis functions of order 6 is used to estimate the functional form. Figures 5.4 it can be seen that the forcing function is the major source of variation rather than the second derivative for both the steps. The problem we try to solve is how we can compare these dynamics. Can we monitor the degree of the evolution of the two steps? We propose two different solutions: a two-step approach and a distance based approach, to compare and estimate the dynamics.

In the two-step approach, once estimated the model of the first set of contours, we propose to fit the data of the second step with the first estimated model. Thus in analogy with regression we build the equation with the first data set and then predict the response for a new one related to the second step.

Once having found coefficients as components of the variability, we pursue a somewhat different approach by introducing a distance among the coefficients of the two models estimated into the two steps. This distance seeks to investigate the modes of variability from the first to the second step of observation.

The effect of introducing a distance among the models in two steps of the observation is that, in evaluating particular curve candidates to be more fast in the evolution, we can quantify the degree of changes.

The twelve functions obtained by estimating the Principal Differential Equation give indication on the dynamic variability, thus we define the following distance among the two models. Let us consider two models Pde_j, $Pde_{j'}$ defined on the same support S:

- model 1: $Pde_j = \{\alpha_x(s), \alpha_y(s), \beta_{1x}(s), \beta_{2x}(s), \beta_{1y}(s), \beta_{2y}(s)\}$,
- model 2: $Pde_{j'} = \{\alpha_{x*}(s), \alpha_{y*}(s), \beta_{1x*}(s), \beta_{2x*}(s), \beta_{1y*}(s), \beta_{2y*}(s)\}$

each of them can be defined by a compound of six functions.

The distance between Pde_j, $Pde_{j'}$ is given by

$$\begin{aligned} d(Pde_j, Pde_{j'}) = & \sqrt{\int_{s\in S} (\alpha_x(s) - \alpha_{x*}(s))^2 ds} + \sqrt{\int_{s\in S} (\alpha_y(s) - \alpha_{y*}(s))^2 ds} + \\ & + \sqrt{\int_{s\in S} (\beta_{1x}(s) - \beta_{1x*}(s))^2 ds} + \sqrt{\int_{s\in S} (\beta_{2x}(s) - \beta_{2x*}(s))^2 ds} + \\ & + \sqrt{\int_{s\in S} (\beta_{1y}(s) - \beta_{1y*}(s))^2 ds} + \sqrt{\int_{s\in S} (\beta_{2y}(s) - \beta_{2y*}(s))^2 ds} \end{aligned} \tag{5.5}$$

This distance allows to quantify the diversity among two consecutive steps. If more than two images are available it takes the dynamic evolution of the distance among the contours.

5.3 Application: a Gioblastoma Multiform study

The early diagnosis of *GBM* is a very important issue in our society, since GBM is the most common malignant histology and represents a disproportionate cause of cancer mortality. The prognosis of GBM remains dismal, with few patients surviving beyond 2 years.

Longitudinal studies indicate a direct relation between the contour function and structural changes in morphological changes. In order to understand the way in which tumor dimension varies, we have their computer tomography scans, which will be transformed into functional data, as explained in the sequel. Then the two approaches proposed in the previous section are applied.

5.3.1 Brain scans processing: from images to contour functions

Captured images using MRI, CT, or any medical image analysis tool are built in different planes. Therefore the resulting images are bi-dimensional. This involves the application of two-dimensional algorithms for image segmentation and corresponding computational methods (Gonzales et al. (2004)[108]). There is a wide variety of semi-automatic segmentation techniques for delineating tumor boundaries in medical images (Barrett and Mortensen (1997)[109] and Boykov and Jolly (2001)[110]). We make the automatic determination of tumor contour based on the observations of the intensity on each pixel forming the image within the region of interest (ROI), so the use of contrast agents on image acquisition, is recommended.

The use of standard ROI templates can reduce the spatial resolution of the study. A small lesion in a large ROI will produce a minor change in the overall result. The result will not be region specific because the precise location of the lesion will be lost. Conversely, a small ROI applied to a large lesion will not reveal the full extent of the lesion. Comparison of equivalent areas in left and right hemispheres by manually drawing ROIs around lesions also is time-consuming, is subjective, can suffer from localisation problems, and can result in erroneous results if diaschisis is present. In our case we choose to use a ROI as a circle form.

The user can choose the center of ROI and the radius of the circle. This input together with the angular resolution provides the number of these vectors. On the direction of these vectors we can calculate the contour points of the tumor. The automatic determination of the contour points is based on the longitudinal resolution (number of points of each vector) which is required to the user. In each vector point we analyse and compare the intensity of the image gray levels. Thus we obtain a number of contour points equal to the number of vectors. These points together with the image are saved and will be used to digitally reconstruct the shape of the tumor by functional data analysis.

The dataset is composed of 15 brain tumor contour functions. The tomography has been done in the same conditions for both the steps of the observations and for all the patients. If there were some differences the images were registered to have the same benchmarks. We thus remove the phase and the amplitude variability as explained in Section 5.1. The data structure observed for longitudinal studies in which functional data are obtained at each of two visits is shown in Figure 5.2.

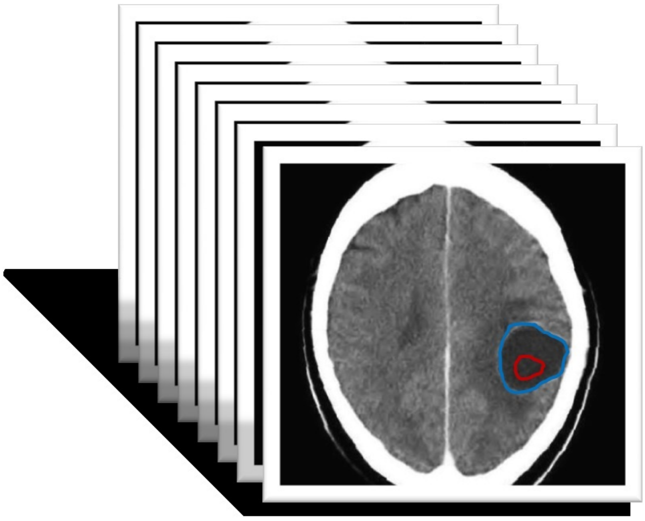


FIGURE 5.2: Tumor boundary from different patients at different times.

The procedure has removed most of the misalignments variation, making it easier to compare curves from different subjects.

The variability captured by the optimal wrapping functions found during this alignment process is shown in Figure 5.3.

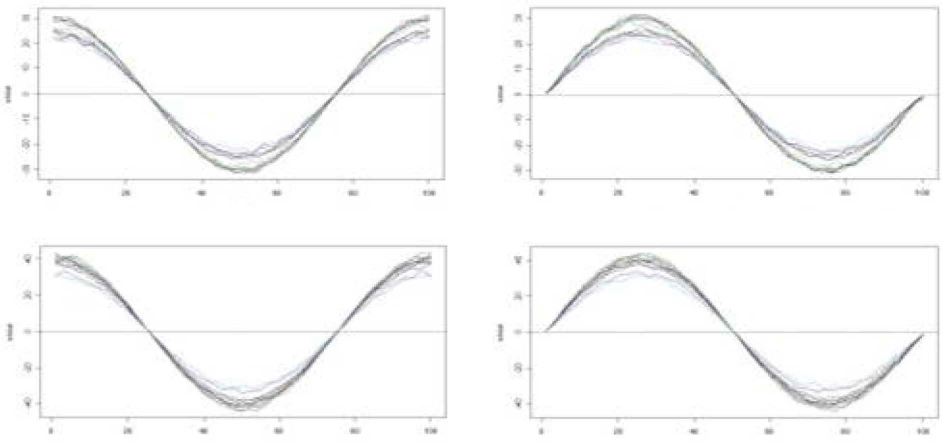


FIGURE 5.3: Registered curves: $X_i(s), i = 1, \dots, 15$ (top left); $Y_i(s), i = 1, \dots, 15$ (top right) $X_i^*(s), i = 1, \dots, 15$(bottom left) $dY_i^*(s), i = 1, \dots, 15$(bottom right)

Here we use functional boxplots to visualize the variability within a single step over the space. Figure 5.4 highlights that there are two contour functions that seem to be anomalous, it means that they have very different shape from the other.

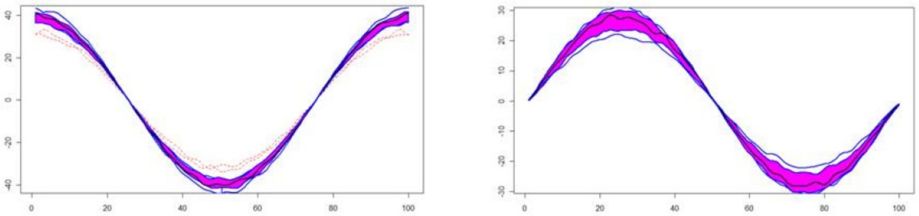


FIGURE 5.4: Functional boxplot for the first step.

There is variability, but the distribution is geometrically symmetric and compact without any significant outliers.

5.4 Results of principal differential analysis on the contour functions

We apply the previous methodology to the contour functions, following the strategies presented in Section 5.2. First, we estimate a structural equation model for the first step. As mentioned before, PDA is calculated for the first step, and then it is applied to the second step.

Residual functions in Figure 5.5 show that the model for the first step is not adequate for the second step since the percentage of explained variability is higher.

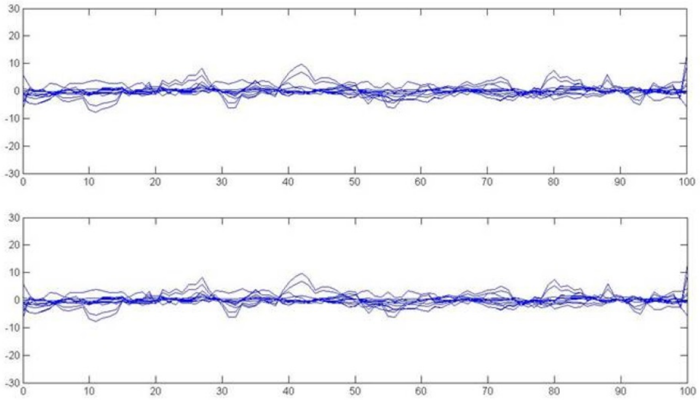


Figure 5.5: Residual function

Following the second approach we apply two PDA on the two stages. Both the components in the two steps yield good fitted values. Points around 0.9 are shown in Table 5.1. The estimated models in the two steps and their explained functional variability can be summarized by their coefficients.

Model components	R^2
X	0.984
Y	0.981
X^*	0.971
Y^*	0.940

Table 5.1: Models fitting, R^2 goodness of fit.

We illustrate in Figure 5.6 the estimated coefficients for the first and second step, related to the the X-component, respectively: $\alpha_x(s)$ (top left), $\beta_{x1}(s)$ (top right) and $\alpha_{x*}(s)$ (bottom left), $\beta_{x*1}(s)$ (bottom right). The pick of the curves shows the zones in which the contour functions increase most rapidly, which indicates that differences between the two steps may be localized to particular tract regions.

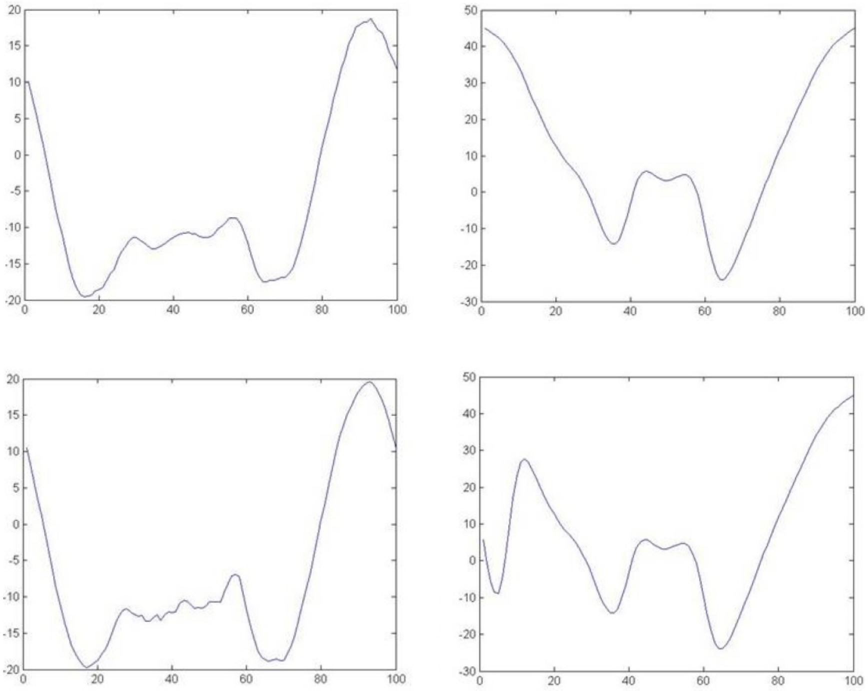


FIGURE 5.6: Estimated coefficient functions for the X component. From top left to bottom right: $\alpha_{x*}(s), \beta_{x*1}(s)$ for the first step; $\alpha_x(s), \beta_{x1}(s)$ for the second step.

The first coefficient, for both the steps (that is the forcing functions), is roughly a mean shift indicating that the overall level of variability varies across subjects over the full domain of the contour functions. The forcing functions can be thus considered as an indicator of large source of variability rather than the coefficients β_{2x} and β_{2x*} that have no impact on the model as can be seen in Figure 5.7.

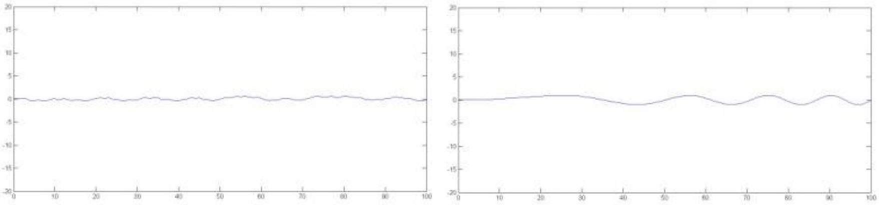


FIGURE 5.7: Estimated coefficient functions $\beta_{2x}(s), \beta_{2x*}(s)$ for the first and second step

We want to investigate the shape variation among the two steps, obtaining a distance among two fitted models. The normalised distance among the coefficient is 0.6, difference due mainly to the forcing component. Despite the general similarities, there are important differences among the contour. It is likely that the first model is a reference model to monitor the variability of the shape and the velocity of the tumor contour propagation has a big weight in controling the model changes.

We deal with contours as continuous functions which better represent the continuous form and the nature of the data. In this case we have no restriction to objects. We have focused our attention to tumor contour functions analysis by the use of FDA. Unlike other papers in the framework of contour functions, we have proposed to use Principal Differential Analysis as a novel application of FDA in image analysis.

The space-varying coefficient function relating velocity to wide margin contour gives summary components. Moreover a linear combination of the contour derivatives gives a prediction of the tumor contour function deformation.

Chapter 6

Software: Prediction of the Dynamic Shape Evolution of Cancer

In the chapters above we have developed new methods and algorithms to predict the evolution in space and time of irregular shapes and their application to particular tumor growth. We implement all these methods in a compact form to offer an ease-to-use program for the interested user.

Taking into account that the methodology used to predict the dynamic shape evolution on cancer is based to mathematical calculus, and the used algorithms require time-consuming mathematical computation, we decided to implement all the methods with Matlab software. In the case of Bayesian prediction we use the R-INLA package for R software in combination with Matlab.

The advances of Matlab are based on the possibility to construct a Graphical User Interface (GUI) and to work with R functions calling, executing and saving results provided by calculus with R scripts. We can also develop our own functions. The specialist can build his own library and call the scripts from the interface. The advantage of this operation consists in the fact that the final user of the program can not modify the script using the interface. So, the motivation to implement these function in a GUI come from the fact that the end-user of this research could be the physicians or the patients who do not have the ability to use a mathematical language and execute Matlab or R scripts.

Requirements engineering for GUI design is a key problem in developing many systems, e.g., operating systems, spreadsheets, search engines, applications and software, etc. To design and build the interface of a software the specialist must take into account some generally accepted principles and technical considerations. In the following we briefly expose some requirements to implement a software interface.

The interface must be “closer” to the final user that in our case can be the physicians to help them to predict the evolution of tumor.

To design the interface we must take into account that the program must be friendly and easy-to-use. In this sense we used familiar GUI patterns (e.g. radio buttons to make a election or decision, push button to execute some commands, edit boxes and sliders to set up a variable between a minimum and maximal accepted values, etc.).

We designed the interface in a way that allows the user to focus on what is most important. The size, color, and placement of each element work together, creating a clear path to understand the interface. A clear hierarchy will play a role in reducing the appearance of complexity (even when the actions themselves are complex). This means that the final user does not necessary know and understand the syntax of used algorithms, they must understand and learn how to use the interface.

The interface makes the connection between user tools and syntax algorithms implemented in background to apply the desired operation to input data and obtain the desired output. As can be seen in the diagram of the software (Figure 6.1) the hierarchy is easy to be understood and follows a natural sense.

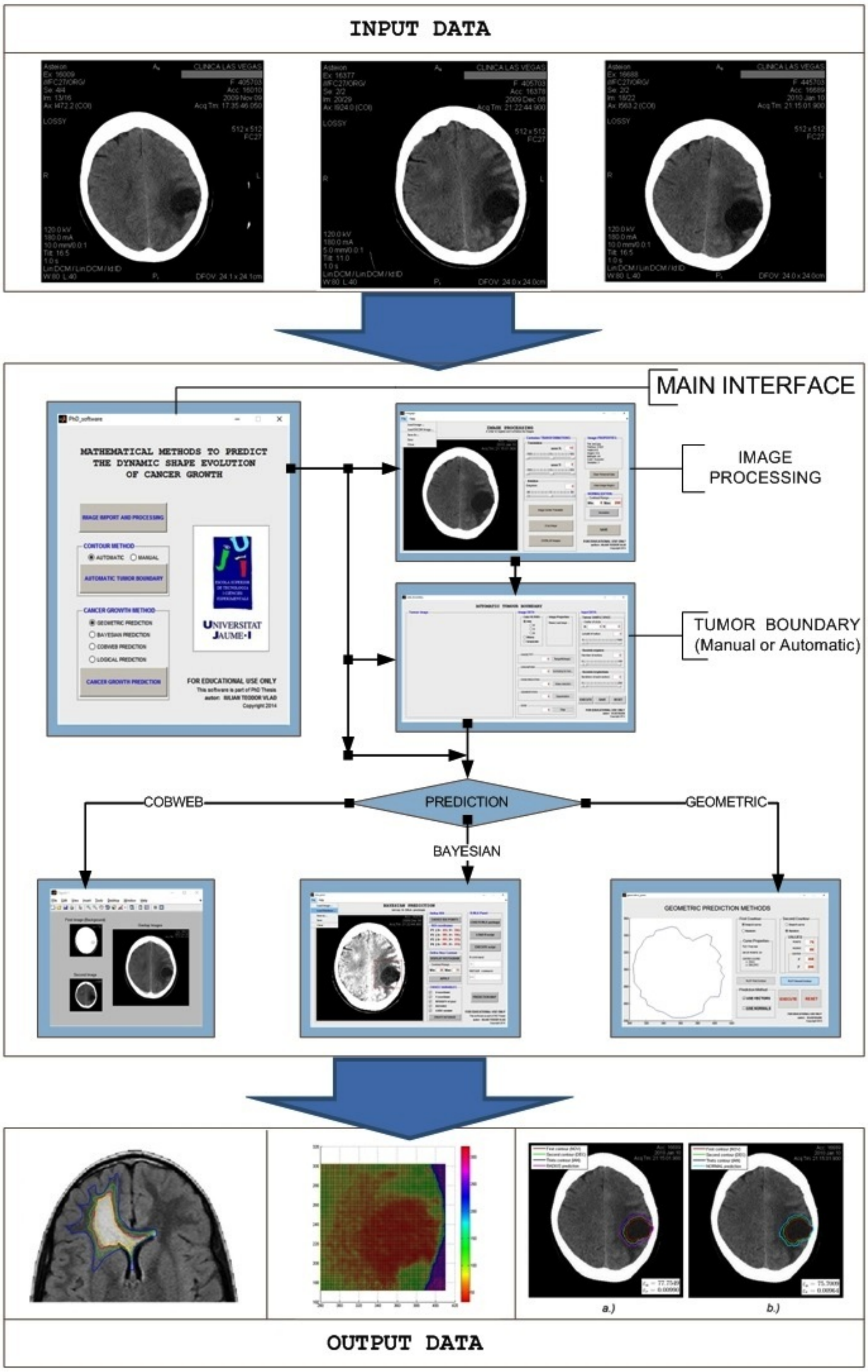
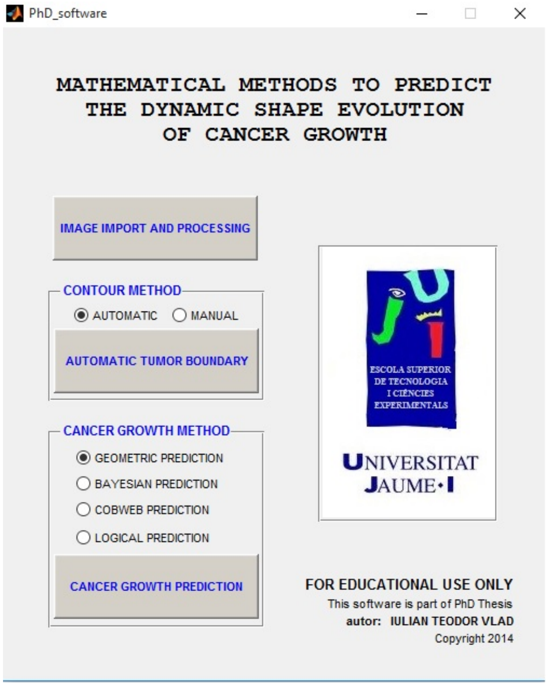

INPUT DATA
MAIN INTERFACE
MATHEMATICAL METHODS TO PREDICT THE DYNAMIC SHAPE EVOLUTION OF CANCER GROWTH
IMAGE PROCESSING
TUMOR BOUNDARY (Manual or Automatic)
PREDICTION
COBWEB
GEOMETRIC
BAYESIAN
a.)
b.)
OUTPUT DATA

FIGURE 6.1: Diagram of PreDySEC

The input are the images (provided by MRI or CT) and the output are the predicted contours or the regions with high probability to evolve in time.

The language, layout, and design are some interface elements that need consistency. A consistent interface enables our users to have a good understanding of how things work, increasing their efficiency.

In accordance with this concept we keep a consistency between different modules. As noted the graphical interface from all modules are quite similar and have similar elements, respect more or less the same design (e.g. the structure of the interface, the used colours, the dimensions and position for buttons and tools, the figures and graphs, etc.).


PhD_software
IMAGE IMPORT AND PROCESSING
AUTOMATIC TUMOR BOUNDARY
CANCER GROWTH PREDICTION
ESCOLA SUPERIOR DE TECNOLOGIA I CIÈNCIES EXPERIMENTALS
UNIVERSITAT JAUME•I

MATHEMATICAL METHODS TO PREDICT
THE DYNAMIC SHAPE EVOLUTION
OF CANCER GROWTH

CONTOUR METHOD
● AUTOMATIC ○ MANUAL

CANCER GROWTH METHOD
● GEOMETRIC PREDICTION
○ BAYESIAN PREDICTION
○ COBWEB PREDICTION
○ LOGICAL PREDICTION

FOR EDUCATIONAL USE ONLY
This software is part of PhD Thesis
autor: IULIAN TEODOR VLAD
Copyright 2014

Figure 6.2: The main interface of PreDySEC

In Figure 6.2 we show the design of the main interface. The objective of the software is to use the information from image analysis and to apply the mathematical models in order to predict the evolution of cancer in a simple and easy way for a normal user. The design is very intuitive and makes the connection between the modules of the package and the mathematical algorithms.

The resources and requirements of the interface are minimal except for the Bayesian prediction module. To use the interface the user needs to have installed Matlab 7 and R software. It can run in a x86 or x64 computer with minimum 2GB of RAM memory, 1.8 GHz of CPU and any operating system. For Bayesian prediction module the execution of the algorithm on such computer are taking approximately two days, so we recommend to use this module on a more powerful computer.

The program is structured into three sections:
- image processing;
- tumor boundary;
- prediction of evolution.

We design the interface for all modules as simple as possible. A modern paradox says that *" it is simpler to create complex interfaces because it is so complex to simplify them"*. We try to not overload the application with unnecessary functions. We provide a clear and concise labels for actions and we try to suggest and guide the user with simple messages.

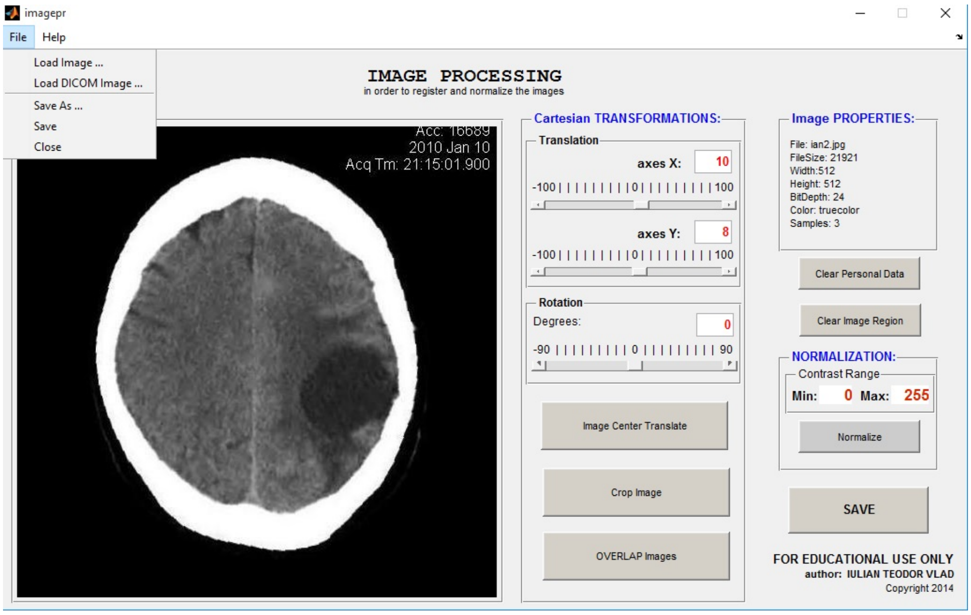
Each component of the interface is designed in such a way to allow the change of information between them. Each module offers the possibility to save the data that can be used by the other module. So the interface offers the user the possibility to remember information from one screen and then use that information on another screen. Starting with data acquisition, passed through each module to obtain the results, the used language clearly respects the proof structure.

In the following we describe the behaviour of each module and we provide step-by-step instruction in order to explain the functionality and advances from one module to another.

6.1 Image processing

To make a prediction about tumor growth we first need data acquisition (at least two images) at predetermined interval time, to see if the tumor is growing . So the first module is designed to preprocess the input data.

The input data consists in a set of images of the tumor taken with CT or MRI at different intervals of time. To be used, all the data must have the same format. Normally these king of images are DICOM format and the most common format are JPEG. The image processing module (see Figure 6.3) offers the user the possibility to convert the DICOM image to JPEG image.



axes X: 10
axes Y: 8
Degrees: 0

File: ian2.jpg
FileSize: 21921
Width:512
Height: 512
BitDepth: 24
Color: truecolor
Samples: 3

Min: 0 Max: 255

Figure 6.3: Image processing module

The user can apply cartesian transformation to the input data in order to register the images and also to normalize the intensity range of each image. For patient identity protection, and to clean the personal information, the user can decide to clear the specific region using the button designed for this.

Once the images are registered and normalized the user can verify the results. To do this he can use the button to overlap the images and see if the images have the same cartesian benchmark. If the results are satisfactory then he can save the images for further use in the next module.

6.2 Tumor contour

Here we provide two ways to perform the contour of the boundary of brain tumor obtained through MRI or CT (automatic or manual). The automatic method is based on interpretation of the intensity of each pixel placed within the ROI. We implement a specific algorithm to determine the brain tumor contour. Each contour defines a set of points that are used in the next section to predict the dynamics of the tumor growing.

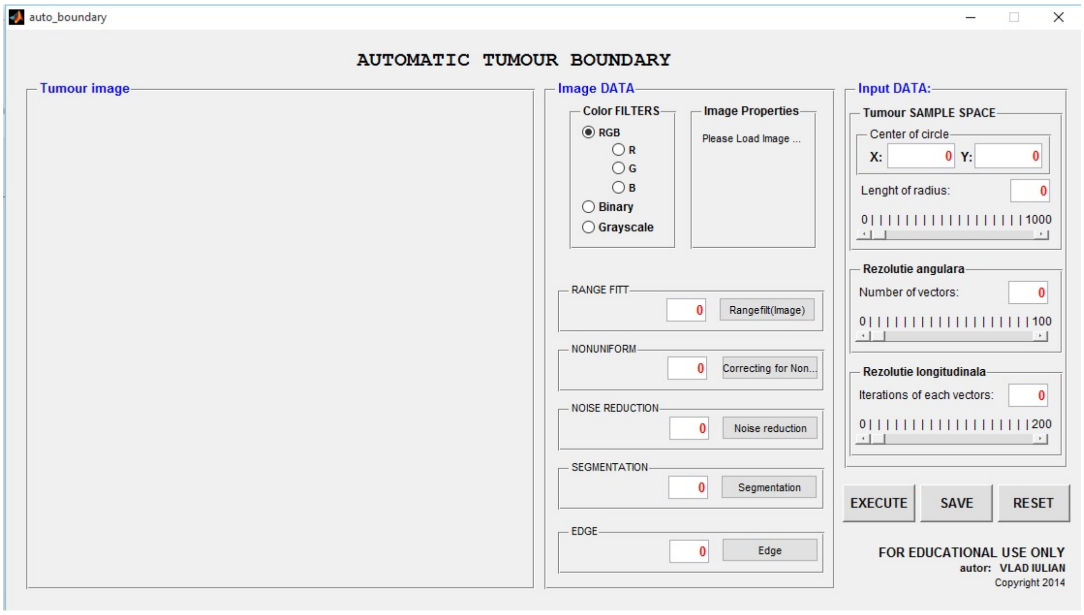


FIGURE 6.4: Automatic tumor boundary

The precision of the automatic method is given mainly by the number of vectors in which the direction of the tumor growth is forecast. They divide the circle counterclockwise in a number of angles equal to the number of vectors (the angular resolution) and by the number of iterations from each vector (the longitudinal resolution). The user must decide about the values of these two resolutions and to define the ROI (the radius and the center of the circle).

The manual method consists in the ability of the user to choose the contour points of the tumor. To be accurate the user can apply to each image different types of filters that can be used to segment the image for better visualization of the tumor.

Once he obtains a good contour he can save the results and pass to the next step to predict the evolution of tumor.

6.3 Prediction of the dynamic shape evolution using the Cobweb algorithm

This module is the most interactive interface. The user is invited to take decision and define different variables. He is guided by message windows. To finalize this step and obtain good results he must understand the requirements and respect the instructions.

The main objective of this code is to implement the cobweb algorithm presented in the second chapter. To perform this, the user is invited to follow some stages.

The first requirement is to determine the approximate tumor center for the first image, and it will be preserved for all other images (second and/or third image if necessary).

The sample space in this situation can be as large as possible, and if the tumor is growing, it could be the entire cranial box. But we can also assume that in a period of time immediately following, the tumor can not grow over certain limits which depend of the structure of the brain.

We can easily construct a sample space which includes the boundary of the future tumor. The choice of the form of this sample space can be directly influenced by the shape and positioning of the brain bulbs. It is a fact that the tumors grow more easily in some directions which depend on the density and the nature of biological material so it is recommended to take into account this information.

Comparison of the two (or three) images taken as input data to determine the rate of increase or decrease of the tumor from each chosen vector (direction) is necessary. In Figure 6.5 we show the capture of the implemented algorithm.

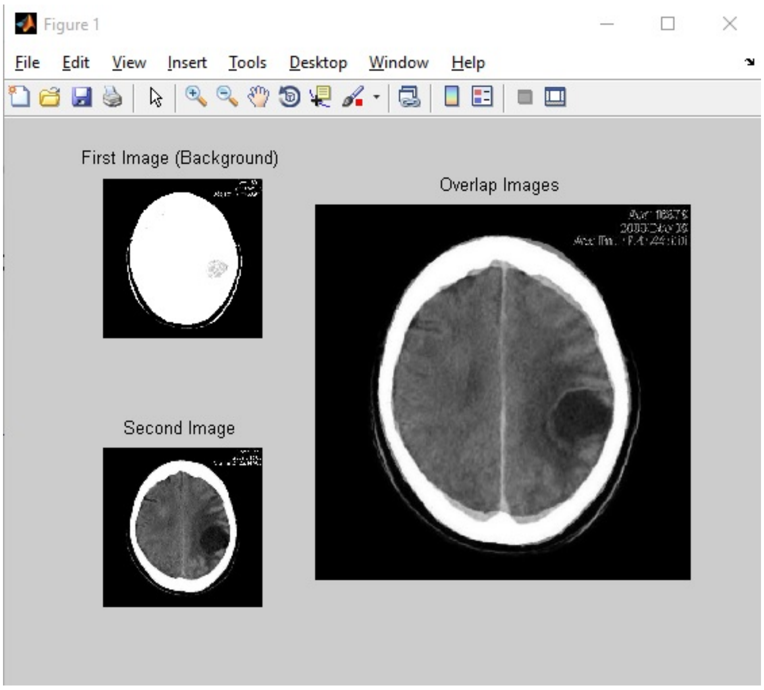


FIGURE 6.5: The prediction with Cobweb module

To get a more precise outline of the tumor prediction, we need to compare the different stages of the tumor development. This will be done after we plot the ROI by entering the coordinate points of the contour for each instance of time. By default the number of points is set to 20 but the user can modify this to increase the precision in detriment of computing time. The user must choose the coordinates of designated points (number of contour points) exactly at the intersection of the line vector with the contour line of the edge of the tumor, and repeat the same procedure for each image.

The output of these functions represent the contour of the predicted tumor after the time designated by the user. This will be plotted in the same two-dimensional plane with the last tumor, together with all stages of the tumor development in time.

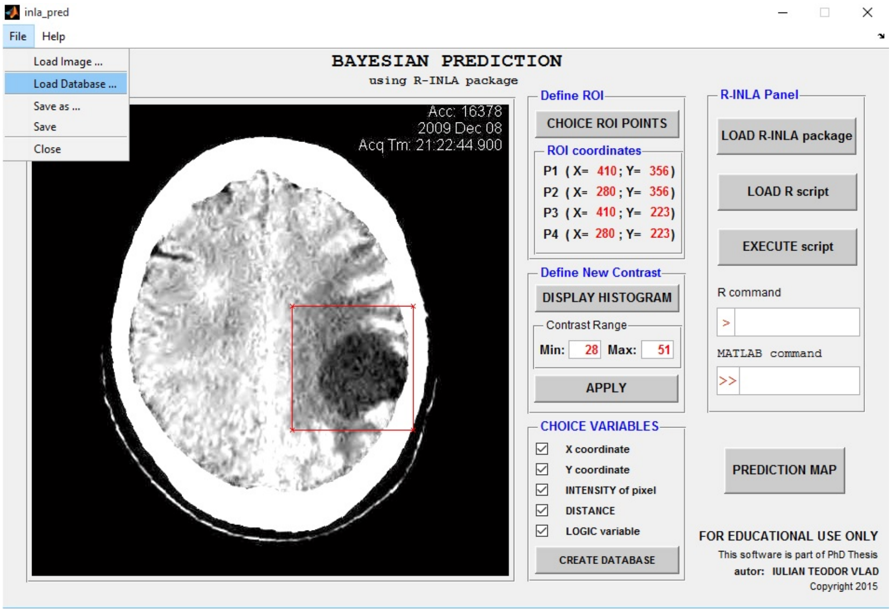
6.4 Bayesian prediction

In this module we use the R-INLA package for R software, so a different type of dataset is needed. We must locate the tumor and extract some information from each image. In order to better locate the cancer tumor we use the image histogram, also named intensity histogram. With this Statistical Image Tool ([68]), the user will be able to judge the entire intensity distribution at a glance.

The horizontal axis of the graph represents the variations in intensity level, while the vertical axis represents the number of pixels in that particular value. The left side of the horizontal axis represents the black pixels, the middle represents medium gray, and the right-hand side represents light and pure white pixels.

This tool also provides the possibility to modify the plotted range so as to show just the intensity level interval of the desired pixels (thresholding). Because the information contained in the graph is a representation of the pixel distribution as a function of the intensity variation, image histograms can be analyzed for peaks and/or valleys which can later be used to determine a threshold value. This threshold value can then be employed for edge detection

Once located the site of the cancer tumor, we then go on to define the Region of Interest (ROI). The ROI must include the boundary of the cancer tumor. Each point of ROI (i.e., each pixel of the cropped image) can be treated as events of a point pattern with a mark given by the intensity level color, which denotes if that event belongs to the cancerous tissue or not.



P1 (X= 410 ; Y= 356)
P2 (X= 280 ; Y= 356)
P3 (X= 410 ; Y= 223)
P4 (X= 280 ; Y= 223)

Min: 28 Max: 51

☑ X coordinate
☑ Y coordinate
☑ INTENSITY of pixel
☑ DISTANCE
☑ LOGIC variable

>

>>

FOR EDUCATIONAL USE ONLY
This software is part of PhD Thesis
autor: IULIAN TEODOR VLAD
Copyright 2015

FIGURE 6.6: Bayesian prediction interface

With the button “choice ROI points” the user draws a squared region by defining the opposite points of the square (see Figure 6.6). To avoid overloading the computer by performing unnecessary calculations, we must choose the ROI that is just enough to contain the tumor and the additional tissue under study, but not so small as to not lose the influence of marginal likelihood.

From the ROI we will extract some information that is needed as input data for further modeling tasks, such as:
- spatial coordinates of each point from ROI;
- at least one covariable (e.g., in our case, since we use images as input data the covariable can be the intensity of each pixel);
- a logical variable zero or one corresponding to the absence or presence of cancer (one if the cell-pixel is cancerous and zero otherwise).

Once the cancer cells have been determined, we can draw the outline of the tumor. The boundaries can be associated with a center of mass and we can compute the distance to each cancer cell.

The data set extracted and used for modeling tasks consists of three matrices $n \times m$ (one from each image), where n represents the number of pixels from ROI, and m are the covariables.

Once we construct the database we can proceed to load the R scripts and execute. The result are saved in the current directory and can be loaded to plot the results. The user have the possibility to plot the data using a Matlab or R graphics.

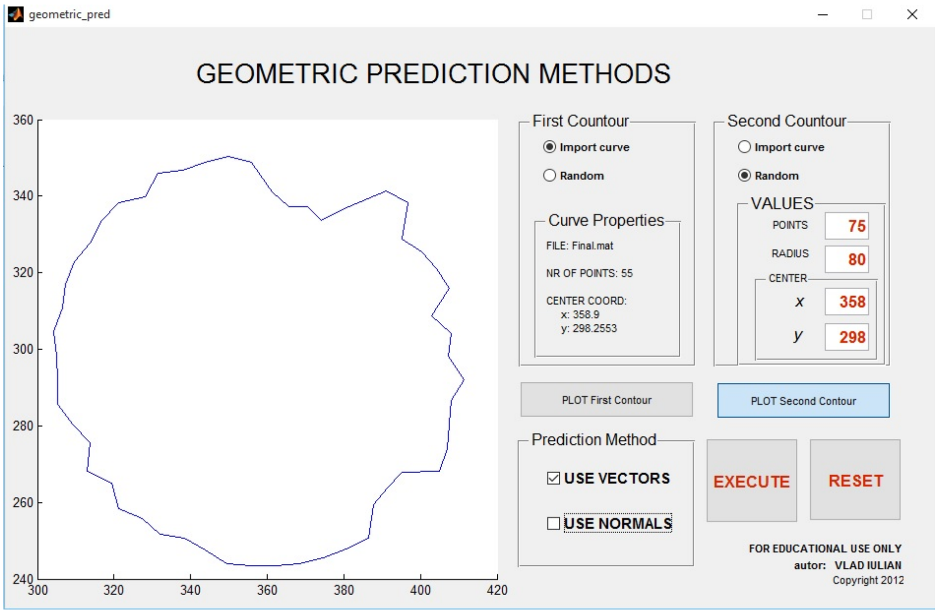
6.5 Geometrical prediction

The predictions with these methods are based on geometric properties of vectors. We propose two methods of geometrical prediction. Both algorithms can also be used to study the evolution in time of any 2D and 3D geometrical forms. One of these methods is based on using the normal vectors of the curvature in each point (normal method) and the other method is based on using the radial vectors from the center of the curvature to the each point (radius method).

All calculations are done using the comparison of contour tumors. Each contour defines a set of points that are used as input of the functional procedure to predict the dynamics of the tumors growth. To perform the prediction, we start from the hypothesis that the velocity of variation in time (the growth) is constant.

The data consists in at least two curves that bound a plane domain provided by the previous interfaces.

In order to obtain a prediction we can also make a simulation with random curves and a verification with a parametric curve. We implement all this mathematical calculations in MATLAB scripts and we build the interface to run both methods (see Figure 6.7).



First Countour
● Import curve
○ Random

Curve Properties
FILE: Final.mat
NR OF POINTS: 55
CENTER COORD:
x: 358.9
y: 298.2553

Second Countour
○ Import curve
● Random

VALUES
POINTS 75
RADIUS 80
CENTER
x 358
y 298

Prediction Method
☑ USE VECTORS
☐ USE NORMALS

FOR EDUCATIONAL USE ONLY
autor: VLAD IULIAN
Copyright 2012

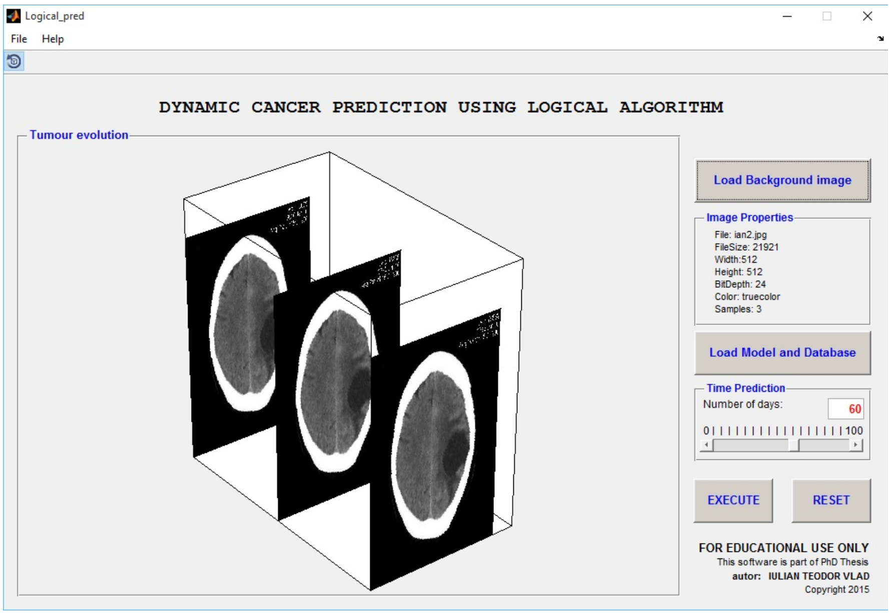
FIGURE 6.7: Geometrical predictions

For simulation we create a MATLAB function to generate two random curves with increase radius and different number of contour points. The dataset was used in Chapter 5.

6.6 Logical prediction in space and time

For this interface we must to develop a new algorithm based on a logical variable. This algorithm can be used to predict in space and time the rising of a new cancer cell based on logical model.

The interface is very intuitive but the background methodology is not. We still work on this model and we hope that soon we can publish the results. In this way we can offer a new method to predict the dynamic of cancer evolution and to compare the result of all these methods. The result will be provided in tridimensional plot and the possibility of rotation is offered.



File: ian2.jpg
FileSize: 21921
Width:512
Height: 512
BitDepth: 24
Color: truecolor
Samples: 3

Number of days: 60

FOR EDUCATIONAL USE ONLY
This software is part of PhD Thesis
autor: IULIAN TEODOR VLAD
Copyright 2015

FIGURE 6.8: Logical prediction module

In Figure 6.8 we implement the graphical user interface of such application.

The communication with the user is very important, so for each module we implement warning message windows that appear when something unexpected is happened and the user introduces wrong information or setups erroneous values from certain variables. We also add a short description of each function in the help menu.

We plan to develop this software and to offer a complete version to interest user as well we have the logical algorithm.

Chapter 7

Conclusions and future research

Mathematical modeling always tries to find a compromise between simplicity of analysis and requirements of realism. On the one hand, we have extremely complex natural and biological systems; on the other hand, we need to formally address some quantitative issues about these systems which can be often done only through the use of mathematical models that may rest on grossly over-simplified assumptions.

On some occasions, a particular mathematical formalism seems to be pre-adapted to a variety of natural and biological systems and can be profitably used to model a diverse set of processes. Double stochastic Cox processes are one class of such models, used here to solve real problems in the field of medicine.

The double stochastic process theory offers a mathematical background to study some natural and physical phenomena in the real world and it takes some conclusions and supplementary information about understanding what is happening in these complex systems.

For most of the realistic problems, the solution of the corresponding exact equation is in practice impossible, so we need to make approximations. Making approximations to solve difficult problems is not a new idea. Appropriate models enable accurate prediction of future behavior, which can be used to control and optimize various aspects of the system in question. However, these approximations are associated with noise induced upon the real problem. The aim is to keep to a minimun this added noise, as this will increase the prediction quality.

An important problem where the use of models to predict the future behavior is essential and important, is the analysis of tumor growth. Modeling the natural growth of tumors is of value in the study of tumor progression, along with that it will be supportive for optimization of screening programs, prognostication, optimal scheduling of chemotherapy and radiation therapy, and assessment of tumor spread (number and size distribution of metastases).

Tumor response to therapy may also be studied by analyzing the effect of therapy on the natural growth of tumor. However, there are mainly two types of growth models for tumors: exponential and non-exponential. According to the Exponential growth model, tumor volume increases exponentially by time. However, studies have shown that tumor growth rate may decline with time which results in nonexponential growth model of tumors. Furthermore, volume doubling time (DT) is the time needed for a tumor to double in volume. DT has been widely used as a quantity for tumor growth rate since its introduction. There are flaws with DT as a quantity for tumor growth rate: the frequency distribution of DT in a population is not normal and there are tumors with very different DT values in a population.

We have presented here a mathematical-statistical approach to analyze the spatio-temporal dynamics of brain tumors. They come in form of processed computer tomography images. We interpret them as collections of image pixels with varying degrees of color intensity levels. As such, they can be considered as a stochastic process, and we make use of spatio-temporal stochastic processes as the right statistical framework.

In addition, in the case of bayesian prediction, we set our modeling strategy within the Bayesian framework which opens doors to model spatial heterogenity, temporal dependence and spatio-temporal interactions amongst the pixels, providing a general modeling framework for such dynamics.

Using this framework, we are able to predict cancer growth in space and time, and show real data analysis. The results are highlighting the regions where the cancer has a higher probability of extending in some future time and shown to be satisfactory.

We have relied upon the Laplace approximation as used in INLA approach. This is a competing methodology with MCMC approaches, and lately a widely used technique, which basically relies on the Matérn spatial interaction structure. If we can not assume this structure on our pixel-based images, we should look for other approaches. A comparison between INLA approach and other existing methods such as spatial logistic regression based on MCMC would be a nice idea.

From the geometric viewpoint we have proposed two methods of tumor growth, where shapes are given as domains bounded by a closed curve in the plane: the radial method (where the planar shape is supposed to be star-shaped) and the normal method (where the shapes estimates at two different occasions are supposed to be similar enough). Both prediction methods proposed here are simple computing, fast, and provide good results. They can help medics cure and better understand the propagation of cancer.

When star-shaped tumors are considered, both methods are accurate in prediction and are computationally fast, and can be used with success in lower computational machines. Comparing the results of the prediction methods with the third image from the set of analysis (boundary of the tumor in January) we note that the radius method is more effective but the normal method is more precise. We must note that the results are directly affected by the precision of contour points and by the resolution of the images.

We should note that we have assumed a constant growth, and in some cases, this growth can not be assumed constant, and we should adapt our modeling strategy to the case of acceleration motion. This is clearly subject of a further research.

For a better adjustment of the prediction models in the very near future, we are trying to develop new models without restricting the growing velocity. In these cases, the function $f(x)$ which transforms the curve α_t into the curve $\alpha_{t+\Delta t}$ is a second-order degree function and from each point of the curve we must solve an equation system in order to find the parameters of each function $f_i(x)$. We hope that this will lead us to remove the predictions error in comparison with real data.

The direction in which current research is going is based on interoperability between different branches of science. It has shown that mixed research teams have much better results. The demands of research topics are thus directed in such way to obtain a more objective results and to the possibility to be used in practice.

Implement into a single interface of all these mathematical prediction methods represent a step forward in this direction. The next step is to implement the logical method and to compare the result of all methods. We have a good feedback and a high interest from the users to use this predictive tool, so we plan to maintain a strong communication with physicians and to develop the PreDySEC software. Once we have funding, we will implement an online platform to support and develop the application.

Appendix A

List of Matlab functions

In this appendix, we present and describe the functions built in Matlab that have been used in this doctoral thesis. This code is available upon contacting the author.

PreDySEC.m: This matlab function represent the source code of the main interface. Here we implement all the action that the code will execute when the user interact with the interface. For example if the user push the button "IMAGE IMPORT AND PROCESSING" the script call and execute the function "imagepr.m" together with "imagepr.fig" and display the module of image processing.

PreDySEC.fig: The image of the PreDySEC interface are designed in this function. Is mean that all the graphical object (images, push buttons, radio buttons, text box, etc.) which user can see on this interface are represented here with his own properties (e.g. color, dimension, position, etc.).

auto_boundary.m: This function load the input image and perform different transformation on the image with the objective to obtain an automatic tumor boundary. The user must define some input variables such a longitudinal resolution, radius, etc. The option to save the results are offered. If the results are not satisfactory the user can choice to manually determine the tumor boundary.

auto_boundary.fig: In this function we have define the interface parameters of the module "Automatic Tumor Boundary".

man_boundary.m: This function is quite similar to "auto_boundary.m" function with the difference that the contour points are defined by the user, clicking on the point. The contour point are saved and based on in we plot the boundary using B-spline.

man_boundary.fig: The interface designed here is dedicated to guide user to manually define the point of the boundary tumor.

imagepr.m: This script contain the commands of processing image function in order to register (translate, rotate, etc.) and normalize the input images. Here we implement the command of each object (axis, uimenu, uipanel, slider, edit box, push button, text box, etc.) from the "Image Processing" interface. The code is executed when the user make some modification on the interface, set the value on edit box or sliders and/or he push button.

imagepr.fig: Contain the objects and characteristics of all these object used to design the "Image Processing" interface.

inla_pred.m: This script call all necessary command implemented to be executed in the case of Bayesian prediction. The source code contain various sections to define and execute different functions such as: define ROI, display histogram. create database, etc.

inla_pred.fig: With this function we design the interface of "Bayesian Prediction" module and we put here the objects that can be used in order to call this procedure.

geom_pred.m: This function is the background script of commands to be executed in order to obtain a geometrical prediction.

geom_pred.fig: With this function we design the interface of "Geometrical Prediction" module. We use the design objects necessary to implement the geometrical prediction methods and to plot the result.

logical_pred.m: Now this function is on a beta version and will have a complete functionality as soon as possible.

logical_pred.fig: The interface of "Logical Prediction" is also in a beta version. We implement this interface maintain the same characteristics of software design.

overlap.m: This script overlap two images. The first represent the background and the second is transparent. Transparency of the second image can be set by modifying the index transparency "AlphaData" between 0 (transparently) and 1 (opaque).

transl.m: Using this script, we are able to translate an image in both directions (x and y axis) in one time. The user must define a reference point (taken as the origin) in the reference image. In the translated image, the center of coordinates will be the point selected by the user. The script calculates the translation matrix between these two points and applies this translation to the second image. The resulting image is saved.

rot.m: This function rotates an image in clockwise direction or backwards if the variable theta (the angle of rotation) is used with negative sign. We apply the command "imrotate" [68] from Matlab, and after the rotation with the given angle we save the new rotated image for further use.

affine_transform.m: This function performs an affine transformation to an image. Taking into account that the images come from a brain tomography, we define an ellipse to superimpose to the reference image, as a reference. The first image represents the background or cover image. The user must choose the eight reference points from the ellipse in both images (first in the reference image, and after in the transformed image). The second image is brought to the same cartesian coordinates with the reference and saved with a different name in the same directory.

newaxes.m: This script plots the new axis x and y which are intersected in the center of the image. The center of the image will be saved and will become the origin axes.

defineROI.m: Using this script, the user can define a region of interest within which the tumor is included. The ROI will be a square or rectangle. To do this we must choose two points: one situated on the top left and the other on the bottom right of the future rectangle.

normalize.m: This function fits the intensity value of each pixel of an image into a particular range. The user can manually set the minimal and maximal values of the given interval and provide compatibility amongst the images.

dicom2jpg.m: This function converts the .dicom (Digital Imaging and Communications in Medicine) image file into a .jpg (Joint Photographic Group) image file. The input argument is the dicom file, and the output is the jpg file saved in the current directory with the same name of the original image.

linie.m: The function of determining the geometric center of the tumor. Is based on the intersection of different lines. The input arguments are the number of lines and return the cartesian coordinates of the intersection point.

dist.m: This script calculates the Euclidean distance between points P and Q with cartesian coordinates (x_P, y_P), (x_Q, y_Q) respectively.

pointsROI.m: The input argument of this function are the study image, the ROI and a minimal and maximal values for the infected pixels. The output data is a $n \times m$ matrix. As rows we have the number of points from ROI and as columns we give the information desired: cartesian coordinates from each point (column 1 contains the x_P coordinates, column 2 contains the y_P coordinates), the intensity value of each pixel in column 3, the logical variable (0 if the cell are cancerous and 1 otherwise) and the distance from center of tumor to each pixel. The logical variable is assigned automatically if the value of intensity pixel belongs to the given interval or not.

verif_points.m: This script plots in the same figure all infected cells in ROI, for all the images, to graphically verify goodness-of-fit of the fitted models.

References

[1] Iulian T. Vlad and Jorge Mateu. A geometric approach to cancer growth prediction based on cox processes. *Journal of Statistics: Advances in Theory and Applications*, 13(1):1–32, (2014). doi: http://dx.doi.org/10.18642/jsata_7100121428.

[2] I. T. Vlad, Ximo Gual, and Jorge Mateu. Two handy geometric prediction methods of cancer growth. *Current Medical Imaging Reviews*, 11(4):254–261, (2015). doi: 10.2174/1573405611666150626173428. URL `www.eurekaselect.com/132483/article`.

[3] Iulian T. Vlad, Pablo Juan, and Jorge Mateu. Bayesian spatio-temporal prediction of cancer dynamics. *Computers and Mathematics with Applications*, 70(5):857 – 868, (2015). ISSN 0898-1221. doi: http://dx.doi.org/10.1016/j.camwa.2015.06.006. URL `http://www.sciencedirect.com/science/article/pii/S0898122115002928`.

[4] E. Romano, J. Mateu, and I. T. Vlad. Principal differential analysis for modeling dynamic contour evolution. a distance-based approach for the analysis of gioblastoma multiform. *Technical Report*, (2014).

[5] D.R. Cox. The theory of stochastic processes. *Champall and hall. London*, (1994).

[6] D.J. Daley and D. Vere-Jones. An introduction to the theory of points processes: Volume i: Theory and methods and volume

ii: General theory and structure. second edition. *Springer*, (2003).

[7] G.E.P. Box, G.M. Jenkins, and G.C. Reinsel. Time series analysis. *Forecasting and Control, Prentice-Hall, Englewood Cliffs*, (1994).

[8] R.J. Bhansali. Autoregressive and window estimates of the inverse correlation function. *Biometrika*, 67:551–566, (1980).

[9] P.J. Brockwell and R. A. Davis. Time series: Theory and methods. *New York: Springer-Verlag*, (1991).

[10] J.D. Hamilton. Time series analysis. *Princeton University Press*, (1994).

[11] B. Postcher and S. Srinivasan. A comparison of order determination procedures for arma models. *Statistica Sinica*, 4:29–50, (1994).

[12] W.A. Fuller. Introduction to statistical time series. *Wiley, New York*, (1976).

[13] A.C. Harvey. Time series models. *John Wiley & Sons. New York*, (1981).

[14] H. Lűtkepohl. Introduction to multiple time series analysis. *Springer Verlag, New York*, (1991).

[15] A. G. Juornel and C.J. Huijbregts. Minning geostatistics. *Academic Press, New York. SUA.*, (1978).

[16] P. Goovaerts. Geostatistics for natural resources evaluation. *Oxford University Press. New York.*, (1997).

[17] J-P Chilés and P. Delfiner. Geostatistics: modeling spatial uncertainty. *Wiley Inter-Science. New York.*, (1999).

[18] N. Cressie. Statistics for spatial data. *John Wiley & Sons, New York.*, (1993).

[19] M.S. Bartlett. The statistical analysis of spatial pattern. *John Wiley & Sons. New York*, (1975).

[20] M.S. Bartlett. A note of spatial pattern. *Biometrics. International Biometric Society*, 20, (1964).

[21] M.S. Bartlett. The spectral analysis of two-dimensional point proceses. *Biometrica Trust. Oxford Journals.*, 51, (1964).

[22] B.D. Ripley. The second-order analysis of stationary point processes. *Journal of Applied Probability*, 13, (1976).

[23] D. Stoyan and H. Stoyan. Fractals, random shapes and point fields.methods of geometrical statistics. *John Wiley & Sons*, (1994).

[24] D. Stoyan, Kendall, S. Wilfrid, Mecke, and Joseph. Stochastic geometry and its applications. series in probability and mathematical statistics. second edition. *John Wiley & Sons*, (1995).

[25] D.R. Brillinger, L.T. Fernholz, and S. Morgenthaler. The practice of data analisys. *Princeton University Press*, (1997).

[26] D.R. Brillinger. Time series: data analysis and theory. *University of California. Holden-Day. SIAM*, (2001).

[27] A.Y Khintchine. Matematical methods in the theory of queueing. *Londra. (Original in Russian (1955)*, (1):1–20, (1960).

[28] D.L. Snyder. Random point processes. *John Wiley & Sons. New York*, (1975).

[29] B.D. Ripley. Modelling spatial patterns (with discussion). *Journal of the Royal Statistical Society.*, B39, (1977).

[30] J. Besag and P.J. Diggle. Simple monte carlo tests for spatial pattern. *Applied Statistics*, 26, (1977).

[31] P.J. Diggle. Statistical analysis of spatial point patterns. *Academic Press. New York*, (1983).

[32] P.J. Diggle, L. Knorr-Held, B. Rowlingson, SuT., P. Hawtin, and T. Bryant. Towards on-line spatial surveillance. in r. brookmeyer and d. stoup, editors, monitoring the health of populations: Statistical methods for public health surveillance. *Oxford University Press*, pages 233–266, (2003).

[33] A.J. Baddeley and B.W. Silverman. A cautionary example on the use of second-order methods for analyzing point patterns. *Applied Statistics*, 40, (1984).

[34] B. Matern. Spatial variation. 2nd edition. *Lecture Notes in Statistics*, (36), (1986).

[35] S. Mase. The treshold method for estimating total rainfall. *Annals of the institute of Statistical Mathematics*, (1996).

[36] M. Schlather, Jr.P.J. Ribiero, and P.J. Diggle. Detecting dependence between marks and locations of marked point processes. *Journal of the Royal Statistical Society*, B 66, (2004).

[37] J. Illian, A. Penttinen, H. Stoyan, and D. Stoyan. Statistical analysis and modelling of spatial point patterns. *John Wiley & Sons. New York*, (2008).

[38] D. Stoyan. Modelling marked point pattern by intensity-marked cos processes. *Statist. and Prob. Letteres*, 78:1184–1199., (2008).

[39] M. Myllymäki. On intensity-dependent markink of spatial log gaussian cox processes. *Workshop on spatial statistics and stochastic geometry. An informal meeting of scientists of Chalmers University of Technology and University of Jyväskylä. Finland*, (2006).

[40] J. Møller, A.R. Syversveen, and R.P. Waagepetersen. Log gaussian cox processes. *Scand. J. Statist*, pages 451–482, (1998).

[41] J. Møller and R.P. Waagepetersen. Statistical inference and simulation for spatial point processes. *Hapman & Hall/CRC, Boca Raton*, (2004).

[42] M. Schlather. On the second-order characteristics of the marked point processes. *Lancaster University. United Kingdom*, (1999).

[43] D.R. Cox and P.A.W. Lewis. The statistical analysis of series of events. *Methuen. London*, (1966).

[44] P.J. Diggle Diggle and A.G. Chetwynd. Second-order analysis of spatial clustering for inhomogeneous population. *Biometrics*, 47, (1991).

[45] Y. Ogata. Seismicity analysis through point-process modeling. *A Review. Pure and Applied Geophysics*, (1999).

[46] E. Choi and Hall. Nonparametric approach to analysis of space-time data on earthquake occurrences. *Journal of Computational and Graphical Statistics*, (8):733–748, (1999).

[47] E. Choi and Hall. On the estimation of poles in intensity functions. *Biometrika.*, 87:251–263, (2000).

[48] S.L. Rathbun. Modeling marked spatio-temporal point patterns. *Bulletin of the International Statistics Institute.*, 55: 379–396, (1993).

[49] S.L. Rathbun and N.A.C. Cressie. A space-time survival point process for a longleaf pine forest in southern georgia. *Journal of the American Statistical Association.*, 89, (1994).

[50] C. Kooperberg and F. OSullivan. Predictive oscillation patterns: A synthesis of methods for spatial-temporal decomposition of random fields. *Journal of the American Statistical Association.*, 91, (1996).

[51] T.C. Haas. Local prediction of a spatio-temporal process with an application to wet sulfate deposition. *Journal of the American Statistical Assoctiation.*, 90, (1995).

[52] S.L. Rathbun. Asymptotic properties of the maximum likelihood estimator for spatio-temporal point processes. *Journal of Statistical Planning and Inference*, 51, (1996).

[53] P.M. Fishman and D.L. Snyder. The statistical analysis of space-time point processes. *IEEE Trans. Inf. Theory*, (1976).

[54] A. Karr. Point processes and their statistical inference. second edition. *Dekker. New York. SUA*, (1991).

[55] M. Bramson and D. Griffeath. On the williams-bjerknes tumor growth model. *Ann. Probab*, pages 173–185, (1981).

[56] N. Cressie. Modelling growth with random sets, in: A. Possolo, C.A. Hayward (Eds.), Spatial Statistics and Imaging, in: IMS Lecture Notes Monogr. Ser. *Hayward, CA.*, 20, (1991).

[57] A.S. Qi, X. Zheng, C.Y. Du, and B.S. An. A cellular automaton model of cancerous growth. *J. Theor. Biol*, 161:1–12, (1993).

[58] T. Lee and R. Cowan. A stochastic tessellation of digital space. *In Mathematical Morphology and Its Applications to Image Processing (J. Serra, ed.) Dordrecht: Kluwer*, pages 217–224, (1994).

[59] A.R. Kansal, S. Torquato, G.R. Harsh, E.A. Chiocca, and T.S. Deisboeck. Simulated brain tumor growth dynamics using a three-dimensional cellular automaton. *J. Theor. Biol*, 203:367–382, (2000).

[60] O. Barndorff-Nielsen and J. Schmiegel. Lévy based spatial temporal modelling, with applications to turbulence. *Russian Math. Surveys*, 59:63–90, (2004).

[61] Kritjana Ý́r Jónsdóttir and Eva B Vedel Jensen. Gaussian radial growth. *Image Anal Sterol.*, 24:117–126, (2005).

[62] Kritjana Ý́r Jónsdóttir and Eva B Vedel Jensen. Lévy-based growth models. *Bernoulli.*, 14(1):62–90, (2008).

[63] D. Richardson. Random growth in a tessellation. *Proc. Cambridge Philos. Soc*, 74:515–528, (1973).

[64] T. Williams and R. Bjerknes. Stochastic model for abnormal clone spread through epithelial basal layer. *Nature J*, 236, (1972).

[65] M. Deijfen. Asymptotic shape in a continuum growth model. *Adv. in Appl. Probab*, 35:303–318, (2003).

[66] N. Cressie and G.M. Laslett. Random set theory and problems of modelling. *SIAM Rev.*, 29:557–574, (1987).

[67] J. Møller. Shot noise cox processes. *Adv. Appl. Prob. (SGSA)*, 35(3):614–640, (2003).

[68] Matlab. Image processing toolbox. Matlab user guide. *Available at.* URL `www.mathworks.co.uk/help/pdf_doc/images/images_tb.pdf`.

[69] E. D. Yorke, Z. Fuks, L. Norton, W. Whitmore, and C. C. Ling. Modeling the development of metastases from primary and locally recurrent tumors: Comparison with a clinical data base for prostatic cancer. *Cancer Research*, 53:2987–2993, (1993).

[70] G. B. West, J. H. Brown, and B. J. Enquist. A general model for ontogenetic growth. *Nature J*, 413:628–631, (2001).

[71] C. Guiot, P.G. Degiorgis, P.P. Delsanto, P. Gabriele, and T.S. Deisboeck. Does tumor growth follow a universal law? *J Theor Biol.*, 225:147–151, (2003).

[72] M.P. Carmo. Differential geometry of curves and surfaces. *Englewood Cliffs, New Jersey, Prentice-Hall*, (1976).

[73] A. Gray. Tubes (second edition). *Birkhauser*, (2004).

[74] A.G. Belyaev, E.V. Anoshkina, S. Yoshizawa, and M. Yano. Polygonal curve evolutions for planar shape modeling and analysis. *Int. Journal of Shape Modeling*, 5:195–217, (1999).

[75] M. Nitzberg, D. Mumford, and T. Shiota. Filtering, segmentation and depth. *Lecture Notes in Computer Science*, 662, (1993).

[76] X. Gual-Arnau and J. Monterde. Curvature approximation from new integral invariants. *Preprint*, (2015).

[77] A. Brú, J.M. Pastor, I. Fernaud, I. Brú, S. Melle S., and C. Berenguer. Super-rough dynamics on tumor growth. *Phys. Rev. Lett.*, 81(18):4008–4011, (1998).

[78] H. Rue and S. Martino. Approximate bayesian inference for hierarchical gaussian markov random field models. *Journal of Statistical Planning and Inference*, 137(10):3177 – 3192, (2007). ISSN 0378-3758. doi: 10.1016/j.jspi.2006.07.016. URL http://www.sciencedirect.com/science/

`article/pii/S0378375807000845`. Special Issue: Bayesian Inference for Stochastic Processes.

[79] H. Rue, S. Martino, and N. Chopin. Approximate bayesian inference for latent gaussian models using integrated nested laplace approximations. (2007).

[80] Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)*, 71(2):319–392, (2009).

[81] F. Lindgren, H. Rue, and J. Lindstrom. An explicit link between gaussian fields and gaussian markov random fields: The SPDE approach (with discussion). *Journal of the Royal Statistical Society, Series B*, 73(4):423–498, (2011).

[82] F. Lindgren and H. Rue. Bayesian spatial and spatiotemporal modelling with r-inla. *Journal of Statistical Software*, (2013).

[83] P. Bremaud. Point processes and queues: Martingale dynamics. *New York. Springer*, (1981).

[84] M. Blangiardo, M. Cameletti, G. Baio, and H. Rue. Spatial and spatio-temporal models with R-INLA. *Spatial and Spatio-temporal Epidemiology*, 4:33–49, (2013).

[85] M. Cameletti, F. Lindgren, D. Simpson, and H. Rue. Spatio-temporal modeling of particulate matter concentration through the SPDE approach. *Advances in Statistical Analysis*, 97(2): 109–131, (2013). doi: 10.1007/s10182-012-0196-3.

[86] D. Simpson, J. Illian, F. Lindgren, S.H. Sorbye, and H. Rue. Going off grid: computationally efficient inference for log-Gaussian Cox processes. *NTNU Technical Report 10, Trondheim University*, (2011). URL http://www.math.ntnu.no/~daniesi/S10-2011.pdf.

[87] R-INLA project. (2012). URL http://www.r-inla.org.

[88] J. L. Harvill. Spatio-temporal processes. *WIREs Computational Statistics*, 2:375–382, (2010). doi: 10.1002/wics.88.

[89] Tilmann Gneiting, William Kleiber, and Martin Schlather. Matrn cross-covariance functions for multivariate random fields. *Journal of the American Statistical Association*, 105(491): 1167–1177, (2010). doi: 10.1198/jasa.2010.tm09420. URL http://dx.doi.org/10.1198/jasa.2010.tm09420.

[90] J. B. Illian and D. K. Hendrichsen. Gibbs point process models with mixed effects. *Environmetrics*, 21:341–353, (2010). doi: 10.1002/env.1008.

[91] J. B. Illian, S. H. Sorbye, and H. Rue. A toolbox for fitting complex spatial point processes models using integreted nested laplace approximations (INLA). *The Annals of Applied Statistics*, 6(4):1499–1530, (2012).

[92] Luke Tierney and Joseph B Kadane. Accurate approximations for posterior moments and marginal densities. *Journal of the american statistical association*, 81(393):82–86, (1986).

[93] R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, (2011).

[94] A. Cosandey-Godin, E.T. Krainski, B. Worm, and J. M. Flemming. Applying bayesian spatiotemporal models to fisheries

bycatch in the canadian arctic. *Canadian Journal of Fisheries and Aquatic Sciences*, 72(999):1–12, (2014).

[95] Daniel P. Simpson, Håvard Rue, Thiago G. Martins, Andrea Riebler, and Sigrunn H. Sørbye. Penalising model component complexity: A principled, practical approach to constructing priors. *NTNU Technical Report 10, Trondheim University*, (2014). URL http://arxiv.org/abs/1403.4630.

[96] David J Spiegelhalter, Nicola G Best, Bradley P Carlin, and Angelika Van Der Linde. Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639, (2002). URL www.maths.usyd.edu.au/u/jormerod/BayesNotes/Papers/SpiegelhalterEtal2002.pdf.

[97] LI Pettit. The conditional predictive ordinate for the normal distribution. *Journal of the Royal Statistical Society B*, 52: 175–184, (1990).

[98] S. Geisser. Predictive inference: an introduction. *Chapman and Hall, London*, (1993).

[99] L. Held, B. Schrödle, and H. Rue. Posterior and cross-validatory predictive checks: a comparison of MCMC and INLA, a chapter in statistical modelling and regression structures: Festschrift in honour of Ludwig Fahrmeir. (2009). URL www.r-inla.org/papers.

[100] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, (2007). doi: 10.1198/016214506000001437. URL http://dx.doi.org/10.1198/016214506000001437.

[101] S. Bauer, R. Wiest, L.P. Nolte, and M. Reyes. A survey of mri-based medical image analysis for brain tumour studies. *Physics in Medicine and Biology*, 58(13), (2013).

[102] A. Hobolth, J. Pedersen, and E Vedel Jensen. A continuous parametric shape model. *Ann I Stat Math*, 55, (2003).

[103] V. V. Kindratenko. On using functions to describe the shape. *Journal of Mathematical Imaging and Vision*, 18, (2003).

[104] I. Epifanio. Functional data analysis in shape analysis. *Computational Statistics and Data Analysis*, 55:2758–2773, (2011).

[105] J. O. Ramsay and B.W. Silverman. Functional data analysis.Second Edition. *Springer-Verlag. NY*, (2005).

[106] E. Romano, J. Mateu, and I. T. Vlad. A functional predictive model for monitoring variation in shapes of brain tumours. *Technical Report*, (2014).

[107] E. Romano, J. Mateu, and I. T. Vlad. Automatic contour detection and functional prediction of brain tumour boundary. *A cura di: Akinori Okada Donatella Vicari Giancarlo Ragozini, Analysis and Modeling of Complex Data in Behavioural and Social Sciences. PADOVA:CLEUP, ISBN: 978-88-6129-916-0, Anacapri, Italy*, (2014).

[108] R.C. Gonzales, R.E. Woods, and S. L. Eddins. Digital image processing using matlab. *Pearson, Prentice Hall, Upper Saddle River, NJ.*, (2004).

[109] W. Barrett and E. Mortensen. Interactive live-wire boundary extraction. *Med Image Anal*, 1:331–341, (1997).

[110] Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in nd images. *In International conference on computer vision*, (2001).

THIS thesis was typeset using LaTeX, the body text is set to 11 points, A4, book, two side page format respecting UJI template and contain 207 pages. A template, which can be used to format a PhD thesis with this look and feel can be found on my page at www3.uji.es/~vlad/.