DEPARTAMENTO DE INTELIGENCIA ARTIFICIAL

Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

PhD THESIS

Directional-linear Bayesian networks and applications in neuroscience

Author

Ignacio Leguey Vitoriano
MS in Mathematical Engineering

PhD supervisors

Pedro Larrañaga
PhD in Computer Science

Concha Bielza
PhD in Computer Science

2018

Thesis Committee

President:

External Member:

Member:

Member:

Secretary:

A Carmen,
porque la mitad de lo que soy,
te lo debo a ti.

Acknowledgements

I really have to thank many people for their support and help in the last years. This dissertation has been a tough work.

I thank my supervisors, Pedro Larrañaga and Concha Bielza, for their guidance.

I would also like to thank Javier DeFelipe and Ruth Benavides-Piccione for introducing me to the fascinating neuroscience field.

I am very grateful to Shogo Kato, Kunio Shimizu, Shogo Mizutaka and Kotaro Kagawa for their hospitality and friendship that made me feel at home during my stay in Tokyo.

I am thankful to Gherardo Varando, Bojan Mihaljevic and the rest of my colleagues at the Computational Intelligence Group for their valuable help, friendship and amazing work environment. I also include Martín Gutiérrez, because he is like a member of the group.

This work has been possible thanks to the financial support of the following projects: Cajal Blue Brain Project (C080020-09), TIN2013-41592-P and TIN2016-79684-P funded by the ministry of Education, Culture and Sport, S2013/ICE-2845-CASI-CAM-CM funded by the Regional Government of Madrid and Human Brain Project funded by the European Union Seventh Framework Programme under grant agreement No. 720270. During the whole PhD period I have been awarded a FPU Spanish Ministry of Education, Culture and Sport Fellowship (FPU13/01941).

Finally, I want to thank my family and friends for their support and advice, always useful and wise. The last and greatest of my gratitudes go for my parents (Santiago and Begoña), my brother (Guille), my grandparents (Santiago and Carmen), Chema and Marta, because they are my team in life's game. This work is dedicated to them.

Abstract

Since the directional nature of certain data present in multiple areas makes traditional statistics ineffective, directional statistics has gained relevancy in the last decades, having special importance in fields such as meteorology, geology, biology or neuroscience. This importance is connected to the development of new technologies that allow obtaining and processing huge amounts of data.

One of the most frequent problems when dealing with any data is uncertainty. In order to work under uncertainty conditions, probabilistic graphical models are a very useful resource. In particular, Bayesian networks combine probability theory with graph theory to provide a powerful data mining tool. In this dissertation, we apply directional statistics techniques in Bayesian networks. We develop Bayesian network models able to deal with data of a directional nature, which we later adapt to address supervised classification problems where the predictor variables are all directional.

Usually, this directional nature data is jointly observed with linear nature data. Several methods have already been used to deal with data from directional and linear nature together. Nevertheless, never in Bayesian networks. Therefore, this problem is also addressed in this dissertation, where we propose a Bayesian network model that allows the use of variables from either directional or linear nature. To do this, we introduce a dependence measure between variables from different nature. This dependence measure is based on the similarity between the joint density function and the product of its marginal density functions. Thus, we use this measure to capture the dependence between directional and linear variables to develop a Bayesian network model with tree-structure.

Neuroscience is another research field that has experimented a great impulse in recent times. The development of new study techniques and advances in microscopy are driving significant advances in this science. These advances require the use of new statistical and computational techniques that allow the data management and data analysis of the results obtained by neuroscientific experiments. In this dissertation we work on the study of neuronal morphology. Despite the numerous advances and scientific investment being made in this area, the structure of the neurons is not known with precision yet. Furthermore, neuronal morphology plays an important role within the functional and computational characteristics of the brain. Hence, making further advances in this field of study can provide relevant information about the brain and the nervous system.

Within the morphology of the neuron, dendrites are responsible for the synaptic reception and the spread of the neuron through the brain. In the study of the dendrites there are measures of discrete, continuous and directional type. Fitting probability distributions to these measures can be complex or even non-existent, so this type of problem represents a modelling challenge.

This dissertation addresses the study of basal dendritic structure in pyramidal neurons. We propose a method to study and model basal dendritic arbors from the branching angles produced by the dendritic split starting from the soma. To do this, we use directional statistics techniques that allow the proper management of directional data (i.e., the bifurcation angles).

Afterwards, we study the behaviour of these angles depending on the type of dendrite from which it originates and the brain layer in which its neuron (its soma) is located.

Going further on neuronal morphology, we also study the pyramidal neuron classification problem into cerebral cortex layers based on their basal dendrites bifurcation angles. To do this, we use the supervised classification Bayesian network models for directional variables developed in this dissertation. Later, we compare the classification accuracy among these directional classification models to evaluate their efficiency. We also compare with random classification.

Resumen

Debido a la naturaleza direccional de ciertos datos presentes en múltiples areas para los que la estadística tradicional es ineficaz, la estadística direccional ha ido ganando relevancia en las últimas décadas, cobrando especial importancia en campos como meteorología, geología, biología o neurociencia. Esta importancia viene ligada al desarrollo de nuevas tecnologías que permiten la obtención y proceso de una elevada cantidad de datos.

Uno de los problemas más recurrentes cuando se trabaja con todo tipo de datos es la incertidumbre. Para trabajar bajo condiciones de incertidumbre, los modelos gráficos probabilísticos son un recurso muy útil. En concreto, las redes Bayesianas combinan teoría de la probabilidad con teoría de grafos para proporcionar una potente herramienta en minería de datos. En esta tesis, aplicamos técnicas de estadística direccional en redes Bayesianas. Desarrollamos modelos de redes Bayesianas capaces de trabajar con datos de naturaleza direccional, que posteriormente adaptamos para aplicar a problemas de clasificación supervisada donde las variables predictoras son todas de dicha naturaleza.

Generalmente, estos datos de naturaleza direccional se encuentran junto a datos de naturaleza lineal. Ya se han desarrollados métodos para trabajar conjuntamente con datos direccionales y lineales, pero nunca en redes Bayesianas. Por lo tanto, también se aborda este problema en esta tesis, donde proponemos un modelo de red Bayesiana que permite tratar variables tanto de naturaleza direccional como lineal. Para ello, proponemos una medida de dependencia entre las variables de diferente naturaleza contenidas en el modelo, basada en la similitud entre su función de densidad conjunta y sus funciones de densidad marginales. De este modo, utilizamos esta medida para capturar la dependencia entre las variables direccionales y lineales para desarrollar un modelo de red Bayesiana con estructura de árbol.

La neurociencia es otro de los campos que ha experimentado un fuerte progreso en los últimos tiempos. El desarrollo de nuevas técnicas de estudio y avances en microscopía están impulsando significativamente el avance de esta ciencia. Estos avances demandan la incorporación de nuevas técnicas estadísticas y computacionales que permitan el manejo y análisis de los datos y resultados obtenidos por los experimentos neurocientíficos. En esta tesis se trabaja en la morfología neuronal, ya que pese a los numerosos avances y la inversión científica que se está realizando en este área, la estructura de las neuronas no se conoce aún con precisión. Además, la morfología neuronal desempeña un importante papel dentro de las carácterísticas funcionales y computacionales del cerebro, de forma que los avances en este campo de estudio pueden aportar valiosa información sobre el cerebro y el sistema nervioso.

Dentro de la morfología de la neurona, las dendritas son las que se encargan de la recepción sináptica y la propagación de la neurona por el cerebro. En el estudio de las dendritas se encuentran medidas de tipo discreto, continuo y direccional. El ajuste de distribuciones de probabilidad a estas medidas puede ser complejo e incluso inexistente, por lo que este tipo de problemas representa un reto en su modelización.

Esta tesis aborda el estudio de la estructura dendrítica basal en neuronas piramidales. Se propone un método para estudiar y modelizar árboles dendríticos basales a partir de los

ángulos de bifurcación producidos por la división de las dendritas partiendo desde el soma. Para ello, se usan técnicas de estadística direccional que permiten el manejo de los datos direccionales (es decir, de los ángulos de bifurcación) adecuadamente. Posteriormente, se estudia el comportamiento de dichos ángulos en función del tipo de dendrita del que provienen y la capa cerebral en la que esta localizada su neurona (su soma).

Ahondando en el estudio de la morfología neuronal, también se estudia el problema de la clasificación de las neuronas piramidales entre las capas de la corteza cerebral con respecto a los ángulos de bifurcación de sus dendritas basales. Para ello, se usan los modelos de redes Bayesianas desarrollados para clasificación supervisada con variables predictoras direccionales desarrollados en esta tesis. Posteriormente, se compara la precisión de clasificación entre estos modelos de clasificación direccional para evaluar su eficiencia. También se compara con la clasificación aleatoria.

Contents

List of Figures

List of Tables

Acronyms

AIC Akaike information criterion

BAM Brain activity map

BBP Blue brain project

BIC Bayesian information criterion

BIGAS BlueGene active storage

BRAIN Brain research through advancing innovative neurotechnologies

bwC bivariate wrapped Cauchy distribution

CBBP Cajal blue brain project

CDF cumulative distribution function

CIQR circular interquartile range

CLMI circular-linear mutual information

CMI circular mutual information

CRAN Comprehensive R archive network

CPT conditional probability table

CSIC Consejo superior de investigaciones científicas

DAG directed acyclic graph

EPFL École Polytechnique Fédérale de Lausanne

GABA γ-amino-butyric acid

GTAN Gaussian tree-augmented naive Bayes

IBM International business machines

IC Instituto Cajal

FET Future and emerging technologies

FPU Formación de profesorado universitario

FSS feature subset selection

FSSJ forward sequential selection and joining

HBP Human brain project

JP Jones-Pewsey

LL log-likelihood

MAP maximum a posteriori

MDL minimum description length

MI mutual information

MIC conditional circular mutual information

MLE maximum likelihood estimation

NB naive Bayes

P14 14-day-old

PGM probabilistic graphical model

S1HL hind limb somatosensory 1 region

SBIC Schwarz Bayesian information criterion

SnB selective naive Bayes

TAN tree-augmented naive Bayes

UPM Universidad Politécnica de Madrid

vM von Mises

wC wrapped Cauchy

wCNB wrapped Cauchy naive Bayes

wCsmNB wrapped Cauchy semi-naive Bayes

wCsNB wrapped Cauchy selective naive Bayes

wCTAN wrapped Cauchy tree-augmented naive Bayes

WDCGG World data centre for greenhouse gases

Part I

INTRODUCTION

Chapter 1

Introduction

Probabilistic graphical models [Koller and Friedman, 2009] and their family of directed acyclic graphs called Bayesian networks [Pearl, 1988] combine graph theory with probability theory to produce a useful tool in data mining. The network structure retains the independence relationships between the variables through conditional probabilities, easily interpretable to find associations between them. The joint probability distribution factorization of a Bayesian network reduces the computational cost for high dimensional distributions. Applying mathematical methods, any type of inference can be conducted. Furthermore, feature selection methods and missing data handling can be performed easily either in the learning or the inference process. Thus, for these reasons among others, Bayesian network models are chosen as a reference paradigm to deal with uncertainty.

Directional statistics [Mardia and Jupp, 2009; Ley and Verdebout, 2017] deals with n-dimensional directions, axes or rotations. Data in the form of angles, day time, weeks, etc. can be also considered directional (i.e., they present a directional nature). Directionality arises in almost every field in science, e.g., in earth sciences as earthquakes, in biology as the animals path, in meteorology as the wind direction, in neuroscience as the direction of axons and neuronal dendrites, in microbiology as the protein dihedral angles, etc.

Circular data refers to information measured in radians and distributed on the circle, while directional data is a more general term referring to directional vectors in an n-dimensional Euclidean space. Its properties do not allow the use of classical statistics.

Despite of their ability to model the relationship between variables, Bayesian networks are hardly developed on directional domains. Directionality in Bayesian networks can be found in simple classifiers for specific directional distributions [López-Cruz et al., 2015]. Indeed, it is difficult to find Bayesian networks that combine variables from different nature, where discrete nature is the most developed area, continuous nature is only used for small networks and nothing for directional. Here, we propose a Bayesian network model that deals with data defined on the circle. Furthermore, we go one step further and present a Bayesian network model that allows the use of linear variables and circular variables together.

Unrevealing brain functioning is an important XXI century scientific challenge in neuroscience. Improvements in modern technology and methodology have enabled a huge increase

on the data acquisition quality, revealing important details of different components of the brain, such as the neuron morphology. Neurons are the most basic unit of the nervous system. The human has about 86 billion neurons in his brain [Herculano-Houzel, 2016], and all of them have different morphology (i.e., there are not two equal neurons). Despite the broad research to reveal the neuron structure by Santiago Ramón y Cajal from the late 1890s, its knowledge is still incomplete. In this dissertation, we intend to contribute to the neuron structure research by shedding light on the dendritic pyramidal neurons structure. In particular, we apply directional statistics techniques together with Bayesian network models to study and model the bifurcation angles produced by the basal dendrite branching in pyramidal neurons. In addition, we extend the circular Bayesian network models to develop several directional Bayesian network-based classification models that capture the interaction between directional variables. These classification models are capable to identify the cortical layer that a neuron comes from, based on its basal dendritic bifurcation angle arrangement.

Chapter outline

This chapter is organized as follows. Section 1.1 presents the main hypotheses and objectives of this dissertation. Then, in Section 1.2 the organization of this manuscript is explained.

1.1 Hypotheses and objectives

The research hypotheses of this dissertation can be stated as the following two main points:

- Directional statistics methods can be applied to build well-behaved Bayesian network models, using these methods to deal with variables in the circular domain instead of these of the traditional statistics.
- Angles and directional measures found in the basal dendritic tree neuronal structure play an important role in neuron morphology. In particular, basal dendritic bifurcation angles can be modelled using directional statistics to predict the cerebral cortex layer where the neuron soma lies.

Based on these hypotheses, the main objectives of this dissertation are:

- To develop the methodology to model a Bayesian network that deals with circular variables.
- To develop Bayesian network-based classification models capable to deal with circular variables.
- To develop a dependence measure between circular and linear variables. In addition, to use this measure to model a Bayesian network model that allows the presence of circular variables as well as linear variables.

- To study and model basal dendritic bifurcations in pyramidal neurons. In particular, to apply directional statistics techniques together with Bayesian networks to identify some of the neuron characteristics, such as their position in the brain (i.e., the layer where the soma is located).

1.2 Document organization

The manuscript includes six parts and nine chapters, organized as follows:

Part I. Introduction

This part presents the dissertation.

- Chapter 1 presents the hypotheses and objectives that motivate this dissertation, as well as the manuscript organization.

Part II. Background

This part includes three chapters that introduce the theory and basic concepts used through this dissertation. The state-of-the-art is discussed within each of these chapters.

- Chapter 2 introduces some basic directional statistics concepts, necessary for dealing with data presented in the form of directions or angles. This chapter is focused on the statistics on the circle: some common circular statistics measures, graphical representations, and some of the best-known circular distributions, i.e., the von Mises distribution, the wrapped Cauchy distribution and the Jones-Pewsey distribution. A list of directional statistics used software is also provided.

- Chapter 3 presents probabilistic graphical models, focused on Bayesian networks as the most important part for this research. This chapter gives an overview of the necessary concepts that are used through this dissertation: learning and inference processes in Bayesian networks, supervised classification Bayesian network models and a description of the used software related to Bayesian networks.

- Chapter 4 provides some basic neuroscience concepts related to the research carried out in this dissertation. This includes a brief introduction to the brain structure and its parts, focused on the neuron functions, types and morphology. Nowadays most remarkable neuroscience projects are also presented.

Part III. Contributions to Bayesian networks and directional statistics

This part includes two chapters that present our proposals in Bayesian networks related to directional statistics techniques.

- Chapter 5 presents four different supervised Bayesian classification algorithms where the predictor variables follow all circular wrapped Cauchy distributions. These are the wrapped Cauchy naive Bayes, the wrapped Cauchy selective naive Bayes, the wrapped Cauchy semi-naive Bayes and the wrapped Cauchy tree-augmented naive Bayes classifiers. Here, synthetic data is used to illustrate, compare and evaluate the classification algorithms, as well as a comparison with the Gaussian tree-augmented naive Bayes classifier.

- Chapter 6 introduces a circular-linear mutual information as a measure of dependence between circular and linear variables. Furthermore, a general dependence measure for circular variables is presented, available for variables that follow any circular distribution and can be expressed in a closed form for a general family of distributions. Using this measure, a circular-linear tree-structured Bayesian network that combines circular and linear variables is presented. Finally, this chapter also presents the evaluation of our proposal, as well as a real-world application in meteorology with public data.

Part IV. Contributions to neuroscience

This part includes two chapters that present our proposals in neuroscience related to directional statistics and Bayesian networks techniques.

- Chapter 7 presents the study of the dendritic branching angles of pyramidal cells across layers to further shed light on the principles that determine their geometric shape. Furthermore, this chapter shows the analysis carried out for this purpose as well as the discussion of the obtained results.

- Chapter 8 shows two application of some of the models developed in Chapter 5. This chapter explains the process to model the bifurcation angles generated by the splitting of the dendritic segments of basal dendritic trees from pyramidal neurons. Furthermore, the models developed in Chapter 5 are used to predict which layer a given pyramidal neuron belongs to. The comparison between the models is also presented.

Part V. Conclusions

This part concludes the dissertation.

- Chapter 9 summarizes the contributions provided in this dissertation and discusses the open issues and future work related to this research. Furthermore, this chapters presents the list of publications and current submissions produced in this dissertation.

Part VI. Appendices

This part includes the appendix.

- Appendix A includes the proofs of the theorems for the Wehrley–Johnson conditionals, the circular mutual information and the circular-linear mutual information, proposed in Chapter 6.

Part II

BACKGROUND

Chapter 2

Directional statistics

2.1 Introduction

Directional data is ubiquitous in science, present in areas such as biology, geology, medicine, oceanography, geophysics or geography [Batschelet, 1981]. Nowadays this kind of data have become specially relevant in geophysics, focused on wind direction to obtain a profitable wind energy utilization, and also in neuroscience measuring the orientation of the neurons and modelling the bifurcation angles produced by the split of the dendritic arbors in order to better comprehend the cerebral functioning. The natural periodicity of directional data is the main difference between directional and non-directional data. This characteristic makes classical statistics methods ineffective for dealing with it. While 0° and 360° are considered the same point in directional data, both are considered different in non-directional data. Thus, directional data analysis is different and more challenging than non-directional data.

Directional statistics [Jammalamadaka and Sengupta, 2001; Mardia and Jupp, 2009; Ley and Verdebout, 2017] is a branch of mathematics that provides the techniques and background to deal with directional data. The foundations of directional statistics arise together with those in more common linear statistics. R. A. Fisher wrote in 1953 [Fisher, 1953]:

"The theory of errors was developed by Gauss primarily in relation to the needs of astronomers and surveyors, making rather accurate angular measurements. Because of this accuracy it was appropriate to develop the theory in relation to an infinite linear continuum, or, as multivariate errors came into view, to a Euclidean space of the required dimensionality. The actual topological framework of such measurements, the surface of a sphere, is ignored in the theory as developed, with a certain gain in simplicity. It is, therefore, of some little mathematical interest to consider how the theory would have had to be developed if the observations under discussion had in fact involved errors so large that the actual topology had had to be taken into account. The question is not, however, entirely academic, for there are in nature vectors with such large natural dispersions."

Directional information can be found in two different ways: circular data and directional data. We talk about circular data for those measures represented by the *compass* or the *clock* [Mardia and Jupp, 2009]. These circular observations are commonly presented as unit

vectors on the circle. There are many situations where the data consist of directions in three dimensions. These data may be represented as points on the sphere, and they are commonly called directional data.

Chapter outline

Section 2.2 explains the techniques for dealing with circular data and reviews some of the best-known circular densities functions such as von Mises, wrapped Cauchy or the Jones-Pewsey family. In Section 2.3 the software for working with directional statistics is briefly presented.

2.2 Statistics on the circle

A circular observation can be regarded as a point on a circle of unit radius or a unit vector in the plane. Once an initial direction and orientation of the circle have been chosen, each circular observation can be defined by the angle from the initial direction to the point on the circle corresponding to the observation. Circular data is commonly measured in degrees, nevertheless, it is sometimes useful to measure in radians by multiplying by $\pi/180$.

A random variable Θ is said to be circular if it is defined in the unit circumference, which domain is $\Omega_\Theta = [-\pi, \pi)$. As previously mentioned, the main characteristic of this data is the periodicity, where $-\pi$ and π are considered the same point. Therefore, due to the specific circular data properties [Jammalamadaka and Sengupta, 2001; Mardia and Jupp, 2009] some special techniques are necessary to deal with circular data, where traditional non-directional statistics are unsuitable. In this section, a basic introduction for the analysis of circular data is presented.

2.2.1 Summary statistics

Likewise for linear domain, it is useful to summarise the data by appropriate descriptive statistics. It turns out that the appropriate way of constructing these statistics for circular data is to regard points on the circle as unit vectors in the plane and then to take polar coordinates of the sample mean of these vectors. Note that angles θ, $\theta \pm 2\pi$, $\theta \pm 4\pi$,.., $\theta \pm 2\pi k$, $k = 1, 2, ..$, are the same point on the circle, therefore the angle to identify a point in the unit circle is not unique. Thus, referring to an angle, we will implicitly mean that its value will be module 2π.

Given $\theta_1, .., \theta_N$ circular values defined in the unit circle with $\theta_i \in [-\pi, \pi)$, $i = 1, .., N$ and unit vectors with Cartesian coordinates $\boldsymbol{x_i} = (\cos(\theta_i), \sin(\theta_i))$, the most popular location measure is the **mean direction** $\bar{\theta}$ of $\theta_1, .., \theta_N$, defined as the direction of the centre of mass $\bar{x}$ of $\boldsymbol{x_1}, .., \boldsymbol{x_N}$, which Cartesian coordinates are $(\bar{C}, \bar{S})$. Hence

$$\bar{\theta} = \arctan\left(\frac{\bar{C}}{\bar{S}}\right), \tag{2.1}$$

with

$$\bar{C} = \sum_{i=1}^{N} \cos(\theta_i), \quad \bar{S} = \sum_{i=1}^{N} \sin(\theta_i).$$

Note that in circular statistics, $\bar{\theta}$ is not defined as in linear domains $(\theta_1 + .. + \theta_N)/N$, as it depends on where the circle is cut.

Another popular location measure is the Fisher **median direction** [Fisher, 1995] ϕ. It is calculated as

$$\hat{\phi} = \arg\min_{\phi} \pi - \sum_{i=1}^{N} |\pi - |\theta_i - \phi||,$$

where $\hat{\phi}$ is the value of the sample $\theta_1, .., \theta_N$ that minimizes the sum of circular distances.

The length of the centre of mass vector $\bar{x}$, called **mean resultant length**, is denoted as $\bar{R}$. It is defined as

$$\bar{R} = \left(\bar{C}^2 + \bar{S}^2\right)^{1/2}.$$

Since $\boldsymbol{x_i}$, $i = 1, .., N$ are unit vectors, $\bar{R} \in [0, 1]$. If the directions of $\theta_1, .., \theta_N$ are widely dispersed then $\bar{R}$ will be almost 0, otherwise if $\theta_1, .., \theta_N$ are tightly clustered then $\bar{R}$ will be almost 1. Therefore, $\bar{R}$ is a measure of data concentration.

To compare circular data with data on the real line it is useful to use dispersion measures. The **circular variance** [Fisher, 1995] $\bar{V}$ is the simplest of these measures. It is defined as

$$\bar{V} = 1 - \bar{R}.$$

Since $\bar{R} \in [0, 1]$, then $\bar{V} \in [0, 1]$ too. Note that some authors (e.g., [Batschelet, 1981]) refer to circular variance as $\bar{V} = 2(1 - \bar{R}) \in [0, 2]$.

2.2.2 Graphical representation of circular data

Graphical representation of data is a way of analysing linear data as well as circular data.

The simplest circular data representation is the **circular raw data plot**. This circular representation plots each observation as a point on the unit circle. Fig. 2.1 compares the representation of circular data in a circular plot against a traditional linear plot. It is easy to appreciate how the linear plot does not reflect the periodicity of the data.

When the data is grouped, there is also an homologous to the linear histogram but for circular data. This is the **rose diagram**, where the frequencies are represented by areas of sectors around the circle instead of bars in the real line. The circumference is divided into sectors of the same arc length and the area of each sector is proportional to the frequency in the corresponding group. Fig. 2.2 shows the comparison of a rose diagram with the corresponding linear histogram, where the latter clearly ignores the periodical nature of the circular data and displays two modes in a "U" shape.

The boxplot [Tukey, 1977] is a simple and flexible graphical tool. It entails the identification of extreme values and outliers in univariate sets. The **circular boxplot** [Abuzaid et al., 2012] also provides that information for circular data. Fig. 2.3.(a) represents the circular

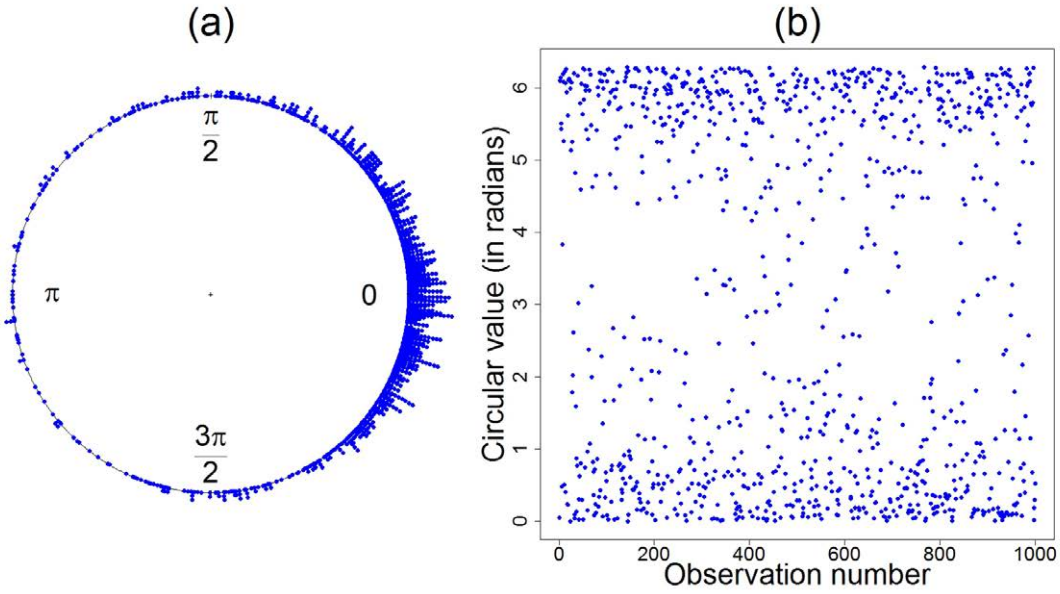

Figure 2.1: (a) Circular plot and (b) linear plot representing the same circular data where the number of instances is 1000.

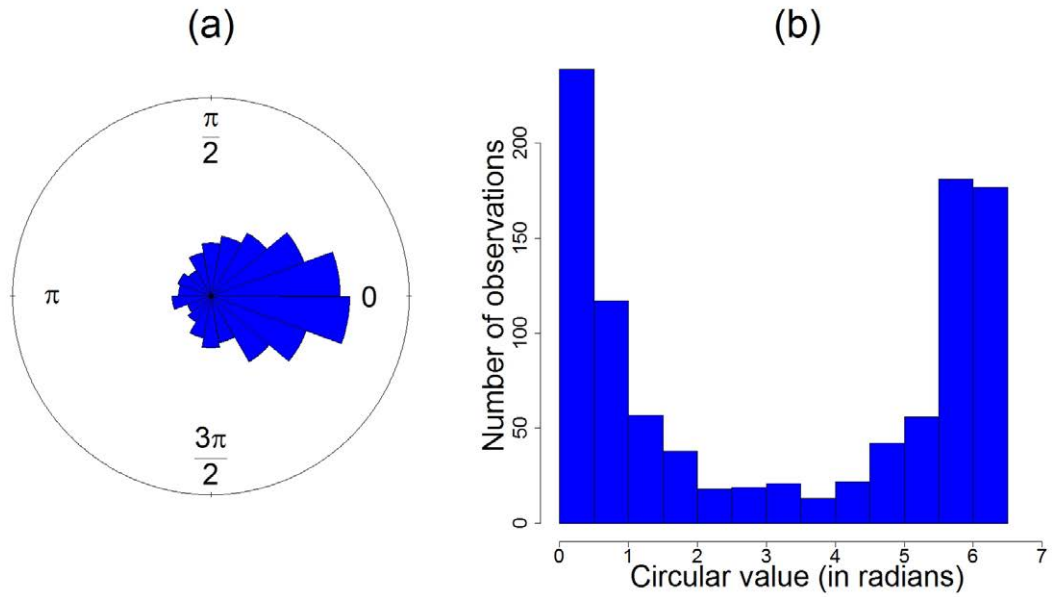

Figure 2.2: (a) Rose diagram and (b) linear histogram representing the same circular data where the number of instances is 1000. The dataset is unimodal and symmetric around 0.

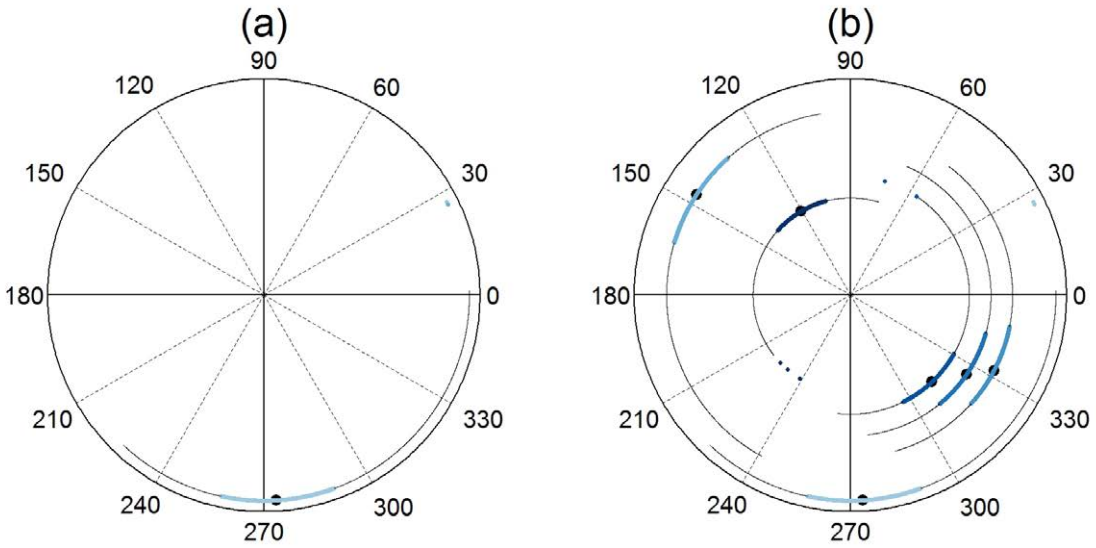

Figure 2.3: (a) Circular boxplot and (b) multiple circular boxplot represented together.

boxplot. The black dot is the median direction, the colored lines are the boxes (from the lower quartile (Q1) to the upper quartile (Q3)), the black lines are the whiskers that depend on the circular interquartile range (CIQR $\equiv$ Q3-Q1) and a concentration parameter of the distribution, and the colored dots are the outliers that do not belong to the box-and-whiskers interval. In addition, as shown in Fig. 2.3.(b), the circular boxplot allows the representation of multiple univariate sets in the same circumference [Leguey et al., 2016b].

2.2.3 Probability density functions

Several probability densities, $f_\Theta(\theta)$, have been used to model circular data. The simplest way to obtain a circular density is by *wrapping*. A random variable X on the real line is wrapped around the circumference of the unit circle to generate a circular random variable Θ, as

$$\Theta = X \mod 2\pi. \tag{2.2}$$

Perhaps, the simplest distribution on the circle, is the **circular uniform**. This distribution is appropriate when no direction is more likely than other. It is obtained by applying Equation (2.2) over a Uniform distribution (i.e., $f(\theta) = 1/2\pi$ for $\theta \in (-\pi, \pi]$). There exist several circular distributions developed by using this method, such as the wrapped Cauchy distribution [Lévy, 1939].

Nevertheless, some special probability densities have been proposed for circular data. The best-known of these is the von Mises distribution [von Mises, 1918], which is an analogous of the Normal distribution in the real line. However, there are more flexible proposals to model circular data. The Jones-Pewsey distribution [Jones and Pewsey, 2005] is a family of symmetric circular distributions where the von Mises distribution and wrapped Cauchy

distribution among others are special cases.

Many other distributions have been proposed in the literature to model circular data, such as the wrapped Normal distribution [de Haas-Lorentz, 2013] or the generalized von Mises distribution [Gatto and Jammalamadaka, 2007] among others [Yfantis and Borgman, 1982; Pewsey, 2008; Kato and Jones, 2010, 2013]. Sections 2.2.3.1 - 2.2.3.3 review the von Mises, wrapped Cauchy and Jones-Pewsey distributions, respectively.

2.2.3.1 The von Mises distribution

The most popular distribution on the circle is the von Mises distribution [von Mises, 1918]. This distribution was introduced by von Mises when studying the deviations of measured atomic weights from integral values. Subsequently Mardia and Jupp [Mardia and Jupp, 2009] proposed five different constructions which lead to it. The von Mises distribution is considered as the analogous of the Normal distribution for linear data, in the literature it is sometimes referred to as *Normal Circular* distribution indeed.

A circular random variable Θ that follows a von Mises distribution, denoted as $vM(\mu, k)$, has density function

$$f(\theta) = \frac{1}{2\pi I_0(k)} e^{k\cos(\theta-\mu)}, \quad \theta \in (-\pi, \pi] \tag{2.3}$$

where $0 < \mu \leq 2\pi$ is the mean direction parameter, $k \geq 0$ is the concentration parameter and

$$I_p(k) = \frac{1}{2\pi}\int_0^{2\pi} \cos(p\theta) e^{k\cos\theta} d\theta$$

is the modified Bessel function of the first kind and order p ($p \in \mathbb{Z}$). When $k = 0$, Equation (2.3) is the circular uniform distribution, otherwise it is unimodal and symmetric about μ. The mode is at $\theta = \mu$ and the antimode is at $\theta = \mu + \pi$. The higher is the k value, the greater is the concentration around the mode. Fig. 2.4 shows the representation of von Mises densities with $\mu = 0$ and different values for the k parameter.

Let $\theta_1, .., \theta_N$ be a random sample from $\Theta \sim vM(\mu_\Theta, k_\Theta)$, defined in Equation (2.3). The maximum likelihood estimators for the parameters μ and k, are the mean direction described in Equation (2.1),

$$\hat{\mu} = \bar{\theta},$$

and $\hat{k} = A^{-1}(\bar{R})$ respectively, where

$$A(\hat{k}_\Theta) = \frac{I_1(\hat{k}_\Theta)}{I_0(\hat{k}_\Theta)} = \bar{R} = \left(\bar{C}^2 + \bar{S}^2\right)^{1/2}.$$

Since the value of $\hat{k}$ cannot be obtained in an exact manner, it has to be approximated

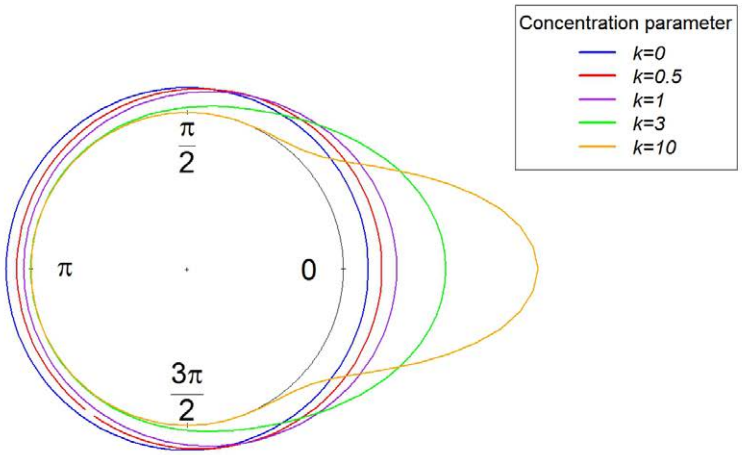

Figure 2.4: The von Mises distribution densities with $\mu = 0$ and $k = 0, 0.5, 1, 3, 10$.

numerically [Sra, 2012]. Fisher [Fisher, 1995] proposed the approximation:

$$\hat{k}_\Theta = \begin{cases} 2\bar{R} + \bar{R}^3 + 5\bar{R}^5/6 & 0 \leq \bar{R} < 0.53 \\ -0.4 + 1.39\bar{R} + 0.43/(1 - \bar{R}) & 0.51 \leq \bar{R} < 0.85 \\ 1/(\bar{R}^3 - 4\bar{R}^2 + 3\bar{R}) & 0.85 \leq \bar{R} \geq 1. \end{cases}$$

2.2.3.2 The wrapped Cauchy distribution

Another of the best-known distributions defined on the circle is the wrapped Cauchy distribution. This was proposed by Lévy [Lévy, 1939] and furthermore studied by Wintner [Wintner, 1947]. It was later obtained by mapping Cauchy distributions onto the circle [McCullagh, 1996] by the transformation $x \mapsto 2\tan^{-1} x$.

A circular random variable Θ that follows a wrapped Cauchy distribution, denoted as $wC(\mu, \varepsilon)$, has density function

$$f(\theta) = \frac{1}{2\pi} \frac{1 - \varepsilon^2}{1 + \varepsilon^2 - 2\varepsilon\cos(\theta - \mu)}, \tag{2.4}$$

where $-\pi \leq \mu < \pi$ is the mean direction parameter and $0 \leq \varepsilon \leq 1$ is the concentration parameter. f in Equation (2.4) is unimodal and symmetric about μ unless $\varepsilon = 0$, which yields the circular uniform distribution. Fig. 2.5 represents the densities of wrapped Cauchy distributions with $\mu = 0$ and $\varepsilon = 0, 0.25, 0.5, 0.75, 0.9$. Further properties of the wC can be found in [Kent and Tyler, 1988] and [McCullagh, 1996].

For parameter estimation of the wrapped Cauchy, the method of moments [Bowman and

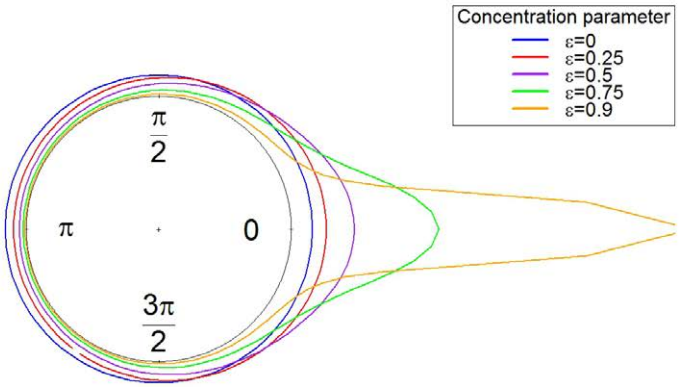

Figure 2.5: The wrapped Cauchy distribution density with $\mu = 0$ and $\varepsilon = 0, 0.25, 0.5, 0.75, 0.9$.

Shenton, 1985] is demonstrated to be more efficient than the maximum likelihood estimators [Kato and Pewsey, 2015].

Let $\theta_1, .., \theta_N$ be a random sample from $\Theta \sim wC(\mu, \varepsilon)$, defined in Equation (2.4). The method of moments-based estimators for parameters μ and ε are

$$\hat{\mu} = arg(\bar{W}), \quad \hat{\varepsilon} = |\bar{W}|,$$

respectively, where

$$\bar{W} = \frac{1}{N} \sum_{j=1}^{N} e^{i\theta_j}.$$

2.2.3.3 The Jones-Pewsey family of distributions

The circular uniform distribution, von Mises distribution and wrapped Cauchy distribution are some of the classical models for directional statistics. These, together with the cardioid [Mardia and Jupp, 2009] and Cartwright power-of-cosine [Cartwright, 1963] distributions are special cases of a wider three-parameter family of distributions on the circle referred to as the Jones-Pewsey family [Jones and Pewsey, 2005].

A circular random variable Θ that follows a Jones-Pewsey distribution, denoted as $JP(\mu, k, \phi)$, has density function

$$f(\theta) = \frac{(\cosh(k\phi) + \sinh(k\phi)cos(\theta - \mu))^{\frac{1}{\phi}}}{2\pi P_{1/\phi}(\cosh(k\phi))}, \tag{2.5}$$

where $-\pi \leq \mu < \pi$ is the location parameter, $k \geq 0$ is the concentration parameter akin to

Table 2.1: The five Jones-Pewsey family of distributions submodels

Submodel	Parameters
Circular uniform	$k = 0$ or $\phi = \pm\infty$ and k=finite
Cardioid	$\phi = 1$
Catwright's power-of-cosine	$\phi > 0$ and $k \to \infty$
wrapped Cauchy	$\phi = -1$
von Mises	$\phi \to 0$

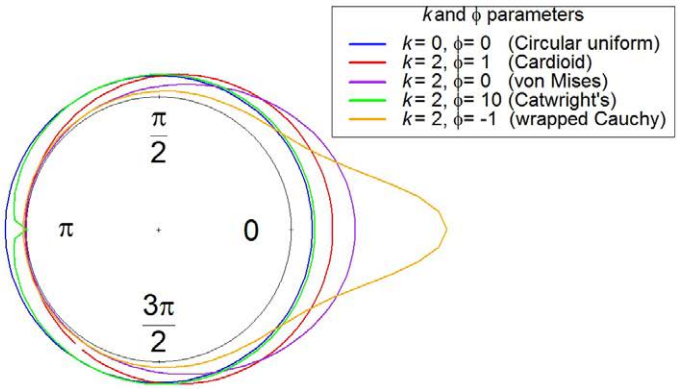

Figure 2.6: Example of Jones-Pewsey distribution densities with $\mu = 0$ and combinations of $k = 0$ with $\phi = 0$ and $k = 2$ with $\phi = -1, 0, 1, 10$.

that in the von Mises distribution, $-\infty < \phi < \infty$ is a shape parameter and $P_{1/\phi}(z)$ is the associated Legendre function of the first kind of degree $1/\phi$ and order 0 [Zwillinger, 1998; Gradshteyn and Ryzhik, 2007]. This family of distributions is symmetric and unimodal on the circle. The five submodels are obtained in the cases presented in Table 2.1.

Fig. 2.6 represents the density of a Jones-Pewsey distribution with $\mu = 0$ and different combinations of k and ϕ. In all cases (but for the circular uniform), it is observable that the densities are unimodal and symmetric around μ.

Let $\theta_1, .., \theta_N$ be a random sample from $\Theta \sim JP(\mu_\Theta, k_\Theta.\phi_\Theta)$, defined in Equation (2.5). Since there are no maximum likelihood estimators for the three parameters, then numerical methods have to be used to approximate them as proposed by [Jones and Pewsey, 2005].

2.3 Software

In this section a brief review of the tools used in this dissertation for working with circular data and directional statistics is given. The software used is the **R** software [R Development

Core Team, 2008], which is a free software environment for statistical computing, graphics and data analysis.

- For basic manipulation and statistical techniques for circular data, `circular` package for **R** [Agostinelli and Lund, 2013] is available at CRAN repository. The content of this package is based on Jammalamadaka and SenGupta book [Jammalamadaka and Sengupta, 2001]. It provides methods for summary statistics, computing, plotting and data testing for non-parametric circular data as well as for different well-known circular distributions such as the von Mises distribution or the wrapped Cauchy distribution, among others.

- The `CircStats` package [Lund and Agostinelli, 2012] is also available at CRAN repository. It is also based on Jammalamadaka and SenGupta book [Jammalamadaka and Sengupta, 2001]. It implements descriptive and inferential statistical analysis of directional data. Also, it includes von Mises distribution and wrapped Cauchy distribution, among others.

- Finally, the book entitled Circular statistics in R [Pewsey et al., 2013] is a useful R programming for circular statistics guide. It provides in-depth treatments of directional statistics. It stresses the use of likelihood-based and computer-intensive approaches to inference and modelling. This book provides a useful revision of some well-known circular and directional distributions such as the von Mises, wrapped Cauchy or Jones-Pewsey, and provides the guidance and the tools to handle with them efficiently in the **R** environment.

Chapter 3

Probabilistic graphical models

3.1 Introduction

Probabilistic graphical models (PGMs) [Koller and Friedman, 2009; Pearl, 1988] are useful tools for data modelling that connect probability theory with graph theory. These models use the graph-based representation to compactly encode a complex distribution over a high-dimensional space. PGMs are composed by two elements: the graphical element and the probabilistic element. In the graphical representation, the nodes correspond to the variables and the edges correspond to the probabilistic interaction between them. The probabilistic element models these probabilistic interactions using conditional probability distributions. The graphical representation can be also seen as the skeleton of the high-dimensional distribution representation. This distribution is split into smaller factors in order to simplify the model. The overall joint distribution is defined by the product of these factors.

Depending on the set of independences that can be encoded and the factorization of the induced distribution, there are two main types of graphical representation of distributions. The first type are called Markov networks, where the used graph is undirected, and the second type are called Bayesian networks, where the graph is directed. In this work, we mainly work with Bayesian networks, as they are more extended for reasoning with uncertainty and several real-world problems have been solved using Bayesian networks [Pourret et al., 2008; Koller and Friedman, 2009].

Chapter outline

Section 3.2 defines useful concepts and notation in order to understand the Bayesian network properties and definitions. Section 3.3 introduces Bayesian networks and how to perform learning and inferences by using them. Extending Bayesian networks as supervised classification models is explained in Section 3.4. In Section 3.5 the software tools used for working with Bayesian networks is briefly presented.

3.2 Useful Bayesian networks concepts

The following concepts are useful for understanding and better comprehend of Bayesian network definitions and their properties.

- A graph $\mathcal{G}$ is a data structure consisting of a set of nodes $\boldsymbol{\mathcal{X}} = \{X_1, .., X_n\}$ and a set of edges $\boldsymbol{\mathcal{E}} = \{(X_i, X_j) | X_i, X_j \in \boldsymbol{\mathcal{X}}\}$ that connect the nodes, where X_i denotes the source node of the edge and X_j denotes the target node of the edge. The edges can be directed or undirected. The latter case ignores the source and target nodes position, since there is no direction.

- A directed acyclic graph (DAG) is a graph $\mathcal{G} = (\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}})$ with only directed edges, called arcs. In addition, the presence of cycles is not allowed, i.e., given the path $\{(X_i, X_j), ..., (X_t, X_s)\}$, it is not allowed that $X_i = X_s$.

- In a DAG, a set of nodes in $\boldsymbol{\mathcal{X}}$ are said to be the parents of $X_j \in \boldsymbol{\mathcal{X}}$, denoted as $\mathbf{Pa}(X_j)$, if the directed arcs from them have the node X_j as the target node, i.e., $\mathbf{Pa}(X_j) = \{X_i | i \neq j, (X_i, X_j) \in \boldsymbol{\mathcal{E}}\}$.

3.3 Bayesian networks

Bayesian networks are based on exploiting conditional independence properties in order to perform a compact representation of the underlying joint probability distribution. A Bayesian network is defined as a pair $\mathcal{B} = (\boldsymbol{\mathcal{G}}, \boldsymbol{\mathcal{P}})$, where $\boldsymbol{\mathcal{G}}$ is the graphical element defined as a DAG, $\mathcal{G} = (\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}})$, and $\boldsymbol{\mathcal{P}}$ represents the probabilistic element, that includes the parameters of the conditional probability functions for each node $X_i, i = 1, .., n$ given the value of its parents $\mathbf{Pa}(X_i) = \mathbf{pa}(x_i)$. Hence, $\boldsymbol{\mathcal{P}} = \left(\boldsymbol{\mathcal{P}}_{X_1|\boldsymbol{Pa}(X_1)}, ..., \boldsymbol{\mathcal{P}}_{X_n|\boldsymbol{Pa}(X_n)}\right)$

According to the $\boldsymbol{\mathcal{G}}$ structure, a Bayesian network encodes in $\boldsymbol{\mathcal{P}}$ the factorization of the joint probability distribution over the variables in $\boldsymbol{\mathcal{X}}$ as:

$$P_{\boldsymbol{\mathcal{X}}}(X_1, ..., X_n) = \prod_{i=1}^{n} P_{X_i|\mathbf{Pa}(X_i)}(x_i|\mathbf{pa}(x_i); \boldsymbol{\mathcal{P}}_{X_i|\boldsymbol{Pa}(X_i)}). \tag{3.1}$$

This factorization avoids the use of high-dimensional probability distributions.

Bayesian networks are efficient probabilistic models with a distinctive property; since the graphical element represents compactly the problem domain, they are easily interpretable.

As an example of a Bayesian network, Fig. 3.1 shows a typical Bayesian network structure. In this example, $\mathcal{B} = (\boldsymbol{\mathcal{G}}, \boldsymbol{\mathcal{P}})$, where $\boldsymbol{\mathcal{G}} = (\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}})$ with $\boldsymbol{\mathcal{X}} = \{X_1, X_2, X_3, X_4, X_5\}$ and $\boldsymbol{\mathcal{E}} = \{(X_1, X_3), (X_2, X_3), (X_2, X_4), (X_3, X_5)\}$, and $\boldsymbol{\mathcal{P}} = \{\boldsymbol{\mathcal{P}}_{X_1|\mathbf{Pa}(X_1)}, \boldsymbol{\mathcal{P}}_{X_2|\mathbf{Pa}(X_2)}, \boldsymbol{\mathcal{P}}_{X_3|\mathbf{Pa}(X_3)}, \boldsymbol{\mathcal{P}}_{X_4|\mathbf{Pa}(X_4)}, \boldsymbol{\mathcal{P}}_{X_5|\mathbf{Pa}(X_5)}\}$. Note that X_1 and X_2 nodes do not have parents, $\mathbf{Pa}(X_3) = \{X_1, X_2\}$, $\mathbf{Pa}(X_4) = \{X_2\}$ and $\mathbf{Pa}(X_5) = \{X_3\}$. Hence, the Bayesian network shown in Fig.

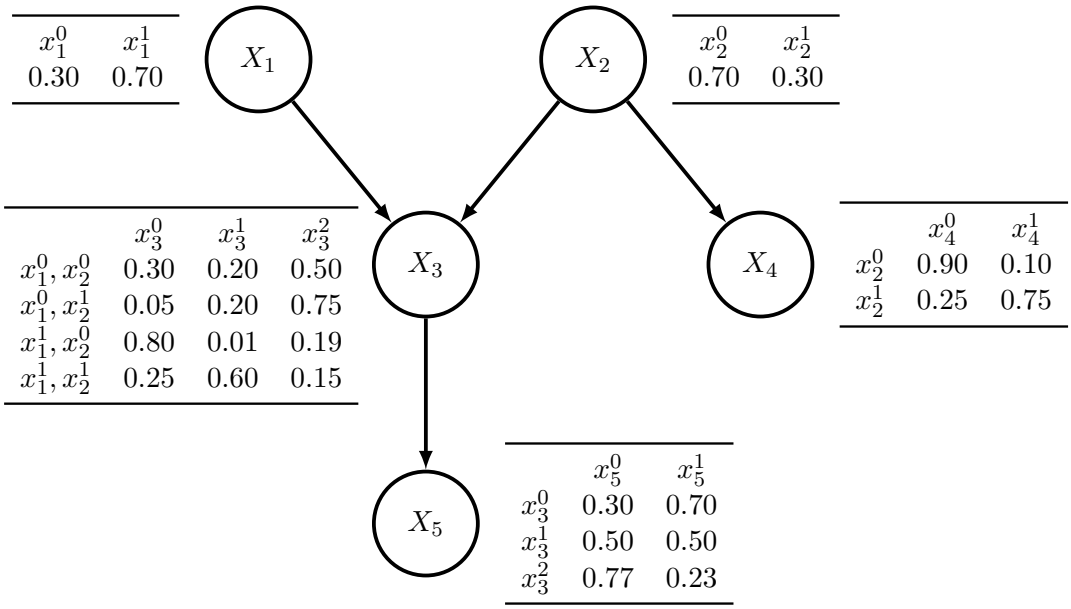

Figure 3.1: Discrete Bayesian network example with five nodes and four arcs. The tables with the probabilistic element are included next to each node. Columns indicate the node value and rows indicate the parents value. The joint probability distribution is shown in Equation (3.2).

3.1 encodes the factorization of the joint probability distribution as:

$$P_{\boldsymbol{\mathcal{X}}}(X_1, ..., X_5) = P_{X_1}(x_1)P_{X_2}(x_2)P_{X_3|X_1,X_2}(x_3|x_1, x_2)P_{X_4|X_2}(x_4|x_2)P_{X_5|X_3}(x_5|x_3). \quad (3.2)$$

3.3.1 Parametrization

Depending on the nature of the variables used in the Bayesian network model, there are discrete Bayesian networks, continuous Bayesian networks and hybrid Bayesian networks. The latter is a combination of continuous and discrete variables. Discrete Bayesian networks and continuous Bayesian networks are briefly presented in the following subsections. There is no further information about hybrid Bayesian networks because they are not used in this dissertation.

3.3.1.1 Discrete Bayesian networks

Discrete Bayesian networks have their variables defined in discrete domains. As shown in Fig. 3.1, for each variable $X_i \in \boldsymbol{\mathcal{X}}$, there is an associated probability distribution for each value $\mathbf{pa}(x_i)$ of its parents $\mathbf{Pa}(X_i)$. The table representation used in Fig. 3.1, called conditional probability table (CPT), is frequently used to display the parameters and probability distribution of each variable given the value of its parents. Let Ω_{X_i} be the possible values that X_i takes, then a CPT consists of the parameters $\mathcal{P}_{ijk} = P_{X_i|\mathbf{Pa}(X_i)}(x_{i_j}|\mathbf{pa}(x_i)_k)$, where x_{i_j} is the

j^{th} value of variable X_i and $\mathbf{pa}(x_i)_k$ is the k^{th} combination of values of the parents of X_i. Hence, the number of parameters in a CPT is the product of the number of possible values of the variable minus one by the number of possible combinations of values of its parents, i.e., $(\|\Omega_{X_i}\| - 1)\|\Omega_{\mathbf{Pa}(X_i)}\|$.

Therefore, the total number of parameters in a discrete Bayesian network is

$$\sum_{i=1}^{n}(\|\Omega_{X_i}\| - 1)\|\Omega_{\mathbf{Pa}(X_i)}\|.$$

3.3.1.2 Continuous Bayesian networks

Continuous Bayesian networks have their variables defined in continuous domains. Gaussian Bayesian networks are the most used, other alternative is to discretize the variables [Fu, 2005].

Discretization approaches

After discretizing the continuous variables, the procedures for the Bayesian network model induction and inference are the same as for discrete Bayesian networks. There are several discretization procedures [see Garcia et al., 2013, for a review]. Nevertheless, often when discretizing a continuous variable, there is a loss of the structure that characterizes it. Furthermore, there are several studies that prove the effect of the discretization in a Bayesian network [Dougherty et al., 1995; Hsu et al., 2000; Yang and Webb, 2003; Hsu et al., 2003; Fu, 2005; Flores et al., 2011a].

Gaussian Bayesian networks

In Gaussian Bayesian networks variables from $\boldsymbol{\mathcal{X}}$ are all Gaussian and have conditional probability distributions that follow Gaussian distributions [Johnson et al., 1970; Wermuth, 1980; Shachter and Kenley, 1989; Tong, 1990; Kotz et al., 2004]. Some interesting properties of the Gaussian assumptions makes this kind of Bayesian networks the most commonly used. Some of these properties are the availability of tractable learning algorithms or the allowance of exact inference [Lauritzen, 1992; Geiger and Heckerman, 1994; Lauritzen and Jensen, 2001], among others. Another important characteristic of the Gaussian Bayesian networks, as explained in [Shachter and Kenley, 1989], is that a Gaussian Bayesian network always define a joint multivariate Gaussian distribution and vice versa. Let Y be a linear Gaussian with parents $\boldsymbol{\mathcal{X}} = \{X_1, .., X_n\}$, that is $f(Y|\boldsymbol{\mathcal{X}}) = \mathcal{N}(\beta_0 + \beta^T\boldsymbol{\mathcal{X}}; \sigma^2)$, where β coefficients are the linear regression coefficients of Y over $\boldsymbol{\mathcal{X}}$. Assuming that $X_1, .., X_n$ are jointly Gaussian and follow $\mathcal{N}(\iota; \Sigma)$, then, the distribution of Y is a Gaussian distribution with $\iota_Y = \beta_0 + \beta^T\iota$ and $\sigma_Y^2 = \sigma^2 + \beta^T\Sigma\beta$. The joint distribution over $\{X_1, .., X_n, Y\}$ is a Gaussian distribution with

$$Cov[X_i; Y] = \sum_{j=1}^{n}\beta_j\Sigma_{i,j}.$$

Therefore, if $\mathcal{B}$ is a Gaussian Bayesian network, then it defines a multivariate Gaussian distribution and vice versa.

It can be seen from Equation (3.1), that the joint probability density of $\{X_1, .., X_n, Y\}$ is given by

$$f(X_1, .., X_n, Y) = \prod_{i=1}^{n} f_{Y|\boldsymbol{\mathcal{X}}}\left(Y|\boldsymbol{\mathcal{X}}; \beta_{0Y|\boldsymbol{\mathcal{X}}}, \beta_{Y|\boldsymbol{\mathcal{X}}}, \sigma^2_{Y|\boldsymbol{\mathcal{X}}}\right).$$

Therefore, the total number of parameters in a Gaussian Bayesian network is

$$2n + \sum_{i=1}^{n}\left(\|\boldsymbol{\mathcal{X}}\| + \frac{\|\boldsymbol{\mathcal{X}}\|(\|\boldsymbol{\mathcal{X}}\| - 1)}{2}\right).$$

Other methods

There are other continuous Bayesian network methods apart from the discretization approaches and the Gaussian assumptions. Some of these different methods do not assume any underlying distribution followed by the variables (i.e., non-parametric methods) [John and Langley, 1995; Hofmann and Tresp, 1996; Bach and Jordan, 2003; Pérez et al., 2009].

Several methods have been used for conditional density estimation in continuous Bayesian networks, such as in Monti and Cooper [1997], where they used neural networks, or in Imoto et al. [2001] and Imoto et al. [2003] where they both used non-parametric regression models.

3.3.2 Learning Bayesian networks

The learning process in a Bayesian network is divided in two steps; the structure learning of the network, and the parameter estimation. These two steps can be addressed in two ways; by expert knowledge [Garthwaite et al., 2005; Flores et al., 2011b], and automatically from a dataset $\mathcal{D}$, when it is available. Both learning ways can be used together, as explained in Heckerman et al. [1995]; Masegosa and Moral [2013]. This dissertation is only focused on learning from data, thus, expert knowledge methods are not reviewed.

3.3.2.1 Structure learning

In a Bayesian network, the associated DAG is called the structure of the network. It has been proven that learning Bayesian network structures is NP-hard [Chickering et al., 1994; Chickering, 1996]. There are three different approaches for structure learning problems: constraint-based criterion, score-search criterion, and hybrid methods, that use both constraint-based and score-search techniques [Koski and Noble, 2012]. The latter is out of the scope of this dissertation, thus it is no reviewed.

Constraint-based

The constraint-based criterion for structure learning of a Bayesian network consists of finding conditional independences between triplets of variables through the use of statistical independence tests. This identifies the edges that are part of skeleton to build the DAG. Once

the undirected graph is built, then the direction of the edges completes the Bayesian network structure. There are several constraint-based methods to find the structure of a Bayesian network [see Spirtes et al., 2000; Koller and Friedman, 2009, among others]. Nevertheless, the best-known is the PC algorithm [Spirtes et al., 2000]. This algorithm (Algorithm 3.1) starts with a complete undirected graph (i.e., edges connecting every pair of nodes) and performs the statistical independence tests in some order to avoid unnecessary calculations. This order is based on the size of the conditional sets of the conditional independence tests. This reduces the number of performed statistical tests and hence, runs faster than other constraint-based algorithms. This algorithm runs in the worst case in exponential time (as a function of the number of variables) and thus it is inefficient when being applied to high dimensional data. Nevertheless, when the true underlying DAG is sparse, which is often a reasonable assumption, this reduces to a polynomial runtime.

Algorithm 3.1 The PC algorithm

1: Given $\boldsymbol{\mathcal{X}} = \{X_1, .., X_n\}$ variables, start with a complete undirected graph on all n variables, with edges between all nodes.
2: For each pair of variables X_i and X_j with $i \neq j$, check if X_i and X_j are independent (i.e., $X_i \perp\!\!\!\perp X_j$); if so, remove the edge between X_i and X_j.
3: For each X_i and X_j that are still connected, and each subset $\boldsymbol{Z}$ of all neighbours of X_i and X_j, check if $X_i \perp\!\!\!\perp X_j | \boldsymbol{Z}$; if so, remove the edge between X_i and X_j.
4: For each X_i and X_j that are still connected, and each subset $\boldsymbol{Z_1}$ of all neighbours and each subset $\boldsymbol{Z_2}$ of all neighbours of $\boldsymbol{Z_1}$, check if $X_i \perp\!\!\!\perp X_j | \boldsymbol{Z_1}, \boldsymbol{Z_2}$; if so, remove the edge between X_i and X_j.
5: ...
6: For each X_i and X_j that are still connected, check if $X_i \perp\!\!\!\perp X_j$ given all the $n-2$ other variables; if so, remove the edge between X_i and X_j.
7: Find colliders (i.e., pair of edges such that they meet in a node) by checking for conditional dependence; orient the edges of colliders.
8: Try to orient undirected edges by consistency with already-oriented edges; do this recursively until no more edges can be oriented.

Score-search

The score-search criterion for structure learning of a Bayesian network consists of tackling the problem as an optimization problem. Heuristic methods are used to find the appropriate structure, and a scoring function is used to evaluate it and leads the searching procedure. The structure with the highest score from among those considered is selected. There are a large number of scoring functions in the literature. All of them have the characteristic of giving a higher score to those networks where the best fitting distribution, given $\mathcal{G}$, is closest to the empirical distribution, with a penalty for the number of parameters.

The *likelihood function* for a graph structure $\mathcal{G}$, given a dataset $\mathcal{D} = \{\boldsymbol{x}^{(i)} = x_1^{(i)}, ..., x_n^{(i)}, i =$

$1, .., N\}$ with $\boldsymbol{\mathcal{X}} = \{X_1, .., X_n\}$ variables and N instances is defined by

$$L(\mathcal{G}; \mathcal{D}) = \prod_{i=1}^{N} \prod_{j=1}^{n} f(x_j^{(i)} | x_{\mathbf{pa}(j)}^{(i)}; \mathcal{G}), \tag{3.3}$$

where $\mathbf{pa}(j)$ represents the indexes of the parents of X_j in $\mathcal{G}$.

Since the formula presented in Equation (3.3) has some practical difficulties (e.g., to specify a large number of parameters), it is useful to work with the *log likelihood* (LL), given by

$$LL(\mathcal{G}; \mathcal{D}) = \sum_{i=1}^{N} \sum_{j=1}^{n} \log f(x_j^{(i)} | x_{\mathbf{pa}(j)}^{(i)}; \mathcal{G}). \tag{3.4}$$

This measure cannot be used as a score function directly, due to the lack of a penalization term in the number of arcs.

The *Bayesian Information Criterion* (BIC) [Schwarz, 1978] is the best-known score function. It uses the LL from Equation (3.4) with a penalization in the number of the parameters:

$$BIC(\mathcal{G}; \mathcal{D}) = LL(\mathcal{G}; \mathcal{D}) - \frac{1}{2} \log(N)|w|,$$

where $|w|$ is the number of required parameters. Alternatively, the negative of BIC is another score function, known as *minimum description length* (MDL) [Rissanen, 1978]: $MDL = -BIC$. Another well-known score function is the *Akaike Information Criterion* (AIC) [Akaike, 1974]. AIC is similar to BIC, but for the penalization term, where N is used instead of $\frac{1}{2} \log(N)$.

The search step explores the space of DAGS and tries to find that with the highest score. The number of possible structures increase more than exponentially with the number of variables. For this reason an exhaustive evaluation sometimes is not suitable. One of the best known search procedures is the $K2$ algorithm [Cooper and Herskovits, 1992], which is summarized in Algorithm 3.2. In this search procedure, considering an ordering over the variables in $\boldsymbol{\mathcal{X}}$, for each node X_i, in the ordering provided, the node from $X_1, .., X_{i-1}$ that most increases the score of the network is added to $\mathbf{Pa}(X_i)$, until no node increases the score or the size of $\mathbf{Pa}(X_i)$ exceeds a predetermined number.

The $K2$ algorithm is an heuristic algorithm. There are other heuristic algorithms for search step. One of the simplest is the *local search* [Hoos and Stützle, 2004]. Let E be a set of eligible changes in the structure and $\Delta(e)$ the change in the score of the network resulting from the modification of $e \in E$. Then, $\Delta(e)$ is evaluated for all e, and the positive change for which $\Delta(e)$ is a maximum is performed. The search finishes when there is no e with a positive value for $\Delta(e)$.

The evolutionary algorithms have become more important in the last decades [see Larrañaga et al., 2013, for a review]. Depending on the space where the searching procedure is performed we distinguish between three different categories: DAG space, ordering space and equivalence

Algorithm 3.2 K2 algorithm

1: Given $\boldsymbol{\mathcal{X}} = \{X_1, .., X_n\}$ nodes, an upper bound u on the number of parents a node may have, and a dataset $\mathcal{D}$.
2: Consideran order for $\boldsymbol{\mathcal{X}}$. Create an empty Bayesian network $\mathcal{B} = (\boldsymbol{\mathcal{G}} = (\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}}), \boldsymbol{\mathcal{P}})$ with $\boldsymbol{\mathcal{E}} = \emptyset$.
3: The score value is set as $Score_{max} = Score(\mathcal{D}, \mathcal{B})$.
4: Following the order for $\boldsymbol{\mathcal{X}}$, for each X_i find $X_j, j = 1, .., i-1$ that maximizes $Score(\mathcal{D}, \mathcal{B}'(\boldsymbol{\mathcal{G}} = (\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}}'), \boldsymbol{\mathcal{P}}'))$, where $\boldsymbol{\mathcal{E}}' = \boldsymbol{\mathcal{E}} \cup (X_j, X_i)$. If $Score_{max} < Score(\mathcal{D}, \mathcal{B}')$, then $Score_{max} = Score(\mathcal{D}, \mathcal{B}')$.
5: Repeat step 4 until $\|\mathbf{Pa}(X_i)\| = u$ or $Score_{max} > Score(\mathcal{D}, \mathcal{B}')$, then go to the next variable.

class space.

The algorithms to search the DAG space consider the learning process by searching in the space of possible DAG structures. Larrañaga et al. [1996c] proposed a genetic algorithm that encodes the connectivity matrix structure in its individuals. In Larrañaga et al. [1996b] they hybridized two versions of a genetic algorithm with a local search operator to obtain better structures. Blanco et al. [2003] demonstrated that using estimation of distribution algorithms (EDAs) leads to comparable or even better results than using genetic algorithms. There are several studies in DAG space algorithm [see Etxeberria et al., 1997; Myers et al., 1999; Wong et al., 1999; Tucker et al., 2001, among others].

The search of the equivalent class space eliminates the redundancy in the DAG space, as demonstrated in [van Dijk and Thierens, 2004]. An evolutionary programming algorithm was also proposed to perform the search in this space [Muruzábal and Cotta, 2004]. They also compared three versions of evolutionary programming algorithms [Cotta and Muruzábal, 2004]. In this space, greedy search seemed to be faster than in the DAG space. Nevertheless, the size of the search space is exponential in the number of variables. van Dijk and Thierens [2004] demonstrated that using an algorithm that consists of hybridizing evolutionary algorithms with local search, improve the results.

To search for the best ordering space (i.e., ordering between the variables) Larrañaga et al. [1996a] used a travelling salesman problem permutation representation with a genetic algorithm. A Bayesian network structure representation composed of *dual* chromosomes was proposed by Lee et al. [2008]. Romero et al. [2004] used two type of EDAs to obtain the best ordering space for the $K2$ algorithm.

All of these types of algorithms are used to capture problem regularities and generate a new solution for searching the best structure in Bayesian networks. Several studies have compared the traditional Bayesian network structure learning algorithms [see Tsamardinos et al., 2006, for an example], and the use of evolutionary algorithms leads to improvements in the computational time and performance [Larrañaga et al., 2013].

3.3.2.2 Parameter estimation

Bayesian network learning process also involves to estimate the parameters $\boldsymbol{\mathcal{P}}$ of the model after the structure $\boldsymbol{\mathcal{G}}$ is fixed. Given a dataset $\mathcal{D}$, there are two ways to fit the parameters: maximum likelihood estimation (MLE) and Bayesian estimation.

MLE consists of finding the parameter set that minimizes the negative log likelihood given by Equation (3.4):

$$\hat{\boldsymbol{\mathcal{P}}} = \arg\min_{\boldsymbol{\mathcal{P}}} -LL(\boldsymbol{\mathcal{B}} = (\boldsymbol{\mathcal{G}}, \boldsymbol{\mathcal{P}}); \mathcal{D})$$

Bayesian estimation consists of estimating the parameters modelled with a random variable $\boldsymbol{\Gamma}$ including prior information encoded in the probability distribution $f_{\boldsymbol{\Gamma}}(\boldsymbol{\mathcal{P}})$ into the problem, and use experience (database) to update the distribution. The problem is based on finding the parameters that maximize the posterior distribution of $\boldsymbol{\Gamma}$ given the database $\mathcal{D}$:

$$\hat{\boldsymbol{\mathcal{P}}} = \arg\max_{\boldsymbol{\mathcal{P}}} f_{\boldsymbol{\Gamma}|\mathcal{D}}(\boldsymbol{\mathcal{P}}|\mathcal{D}).$$

3.3.3 Inference

One of the most interesting Bayesian network properties is the ability to modelling and reasoning in domains with uncertainty. Therefore, Bayesian networks are well designed to answer probabilistic queries. Typically, the Bayesian network will provide some evidence, that is, some of the variables will be instantiated, and the aim is to infer the probability distribution of some other variables.

Given a Bayesian network $\mathcal{B}$ with structure $\boldsymbol{\mathcal{G}} = (\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}})$ fixed, the most common query type is the conditional probability $P(X_q|X_e)$, where $X_e \in \boldsymbol{\mathcal{X}}$ represents the variables that provide some evidence and $X_q \in \boldsymbol{\mathcal{X}}$ are the queried variables. For this type of inference problem, evidence propagation is the most extended method, computing

$$P(X_q|X_e) = \frac{P(X_q, X_e)}{P(X_e)}.$$

This inference process has been proved to be NP-hard [Cooper, 1990; Dagum and Luby, 1993] in the worst scenario case (which is not common).

There are two types of inference: exact and approximate. Exact inference is based on computing analytically the conditional probability distribution over the variables of interest and can be performed in polynomial time when the Bayesian network structure is a polytree [Good, 1961; Kim and Pearl, 1983; Pearl, 1986, 1988]. Otherwise several approaches have been proposed in the literature [Shachter, 1986, 1988; Shachter and Kenley, 1989; Suermondt and Cooper, 1990; Jensen et al., 1990a,b; Suermondt and Cooper, 1991; Díez, 1996; Park and Darwiche, 2003; Darwiche, 2003, etc]. Unfortunately, sometimes inference on complex Bayesian networks may be still infeasible and some approximation techniques based on statistical sampling are used to approximate the result. This is approximate inference. These algorithms provide results in shorter time, albeit inexact. Some of the methods are based on Monte Carlo simulations [see Hernandez et al., 1998; Lemmer and Kanal, 1988, for an

example], and others rely on deterministic procedures [see Bouckaert et al., 1996; Cano et al., 2011, among others].

3.4 Bayesian networks classifiers

Bayesian network classifiers [Friedman et al., 1997] are special types of Bayesian networks designed for classification problems. Supervised classification [Duda et al., 2001] deals with the problem of assigning a label to an instance, based on a set of variables that characterize it. Bayesian network classifiers have several advantages over other classification models, some of these are that they offer an explicit, graphical and interpretable representation of uncertain knowledge, decision theory is naturally applicable for dealing with cost-sensitive problems, they can easily accommodate feature selection methods and handle missing data in both learning and inference phases, etc. [see Bielza and Larrañaga, 2014].

3.4.1 Learning Bayesian network classifiers

Let $\boldsymbol{\mathcal{X}} = \{X_1, ..., X_n\}$ be a vector of features, and C be a class variable. Given a simple random sample $\mathcal{D} = \{(x_1^{(1)}, ..., x_n^{(1)}, c^{(1)}), ..., (x_1^{(N)}, ..., x_n^{(N)}, c^{(N)})\}$, of size N, the Bayesian network classifier structure encodes the conditional independences between the variables $X_1, .., X_n, C$. To assign a label $c^* \in C$ to a new instance $(x_1^*, ..., x_n^*)$ a maximum a posteriori decision rule is used to assign the maximum a posteriori (MAP) label to it:

$$c^* = \arg\max_c P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*, .., .x_n^*) = \arg\max_c P_C(c) P_{\boldsymbol{\mathcal{X}}|C}(x_1^*, ..., x_n^*|c), \tag{3.5}$$

where $P_{\boldsymbol{\mathcal{X}}|C}(x_1^*, ..x_n^*|c)$ factorizes according to the Bayesian network classifier structure, as in Equation (3.1).

Most works in Bayesian network classifiers are mainly focused on discrete domains for the predictive variables. Nevertheless, Bayesian networks with continuous variables have been also studied [Yang and Webb, 2002; Pérez et al., 2006; Flores et al., 2009].

3.4.1.1 Structure learning

Depending on the network structure there are different Bayesian network classifiers. The simplest classifier is the *naive Bayes* (NB) classifier [Minsky, 1961]. An example of its structure with five predictive variables is shown in Fig. (3.2). This classification model assumes conditional independence between the predictive variables given the class, transforming Equation (3.5) into

$$c^* = \arg\max_c P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*, ..x_n^*) = \arg\max_c P_C(c) \prod_{i=1}^{n} P_{X_i|C}(x_i^*|c). \tag{3.6}$$

This assumption is useful when n is high and/or N is small, making $P_{\boldsymbol{\mathcal{X}}|C}(x_1^*, ..x_n^*|c)$ difficult to estimate.

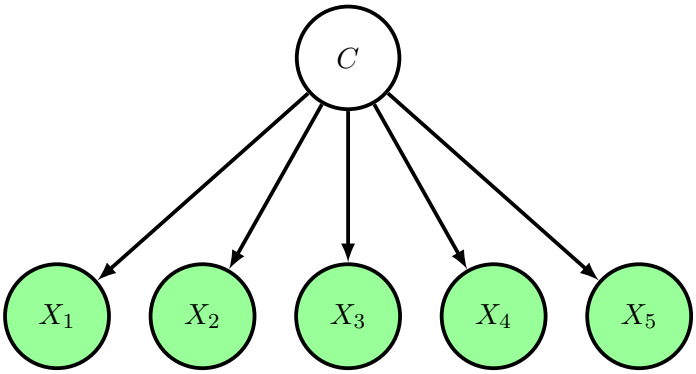

Figure 3.2: Naive Bayes classifier structure with five nodes, from which $P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*,..,x_5^*) \propto P_C(c)P_{X_1|C}(x_1^*|c)P_{X_2|C}(x_2^*|c)P_{X_3|C}(x_3^*|c)P_{X_4|C}(x_4^*|c)P_{X_5|C}(x_5^*|c)$.

The classification performance of the *naive Bayes* classifier could be improved if only non-redundant variables are selected to build the model. *Feature subset selection* (FSS) techniques [Saeys et al., 2007] makes this possible in the so-called *selective naive Bayes* (SnB) classifier. An example of its structure is shown in Fig. (3.3). This model works with a subset $\boldsymbol{\mathcal{X}}_S \in \boldsymbol{\mathcal{X}}$ with $S \subseteq \{1,..,n\}$, that contains the selected features, turning Equation (3.6) into

$$c^* = \arg\max_c P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*,..x_n^*) \propto \arg\max_c P_{C|\boldsymbol{\mathcal{X}_S}}(c|x_1^*,..x_n^*) = \arg\max_c P_C(c)\prod_{i\in S} P_{X_i|C}(x_i^*|c).$$

These FSS requires to consider 2^n structures. Therefore heuristic approaches are used for this search. It may be used a *filter* approach to perform feature selection prior to building the classifier, or a *wrapper* approach is used to build the model by using the classification performance [Saeys et al., 2007]. For the *filter* approach, the most used method consists of scoring the variables through the mutual information (MI) between each feature and the class variable [Pazzani and Billsus, 1997]. Given a pair of discrete variables X_i and X_j, the MI between them is defined as

$$\mathrm{MI}(X_i, X_j) = \sum_{x_i\in X_i}\sum_{x_j\in X_j} p(x_i,x_j)\log\left(\frac{p(x_i,x_j)}{p(x_i)p(x_j)}\right),$$

where $p(x_i, x_j)$ is the joint probability function of X_i and X_j, and $p(x_i)$ and $p(x_j)$ are the marginal probability distributions of X_i and X_j respectively. When both variables are defined in continuous domain, the MI is given by

$$\mathrm{MI}(X_i, X_j) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x_i,x_j)\log\left(\frac{f(x_i,x_j)}{f(x_i)f(x_j)}\right),$$

where $f(x_i, x_j)$ is the joint density function of X_i and X_j, and $f(x_i)$ and $f(x_j)$ are the marginal probability density functions of X_i and X_j respectively.

The *wrapper* approach outputs the feature subset with a higher computational cost since

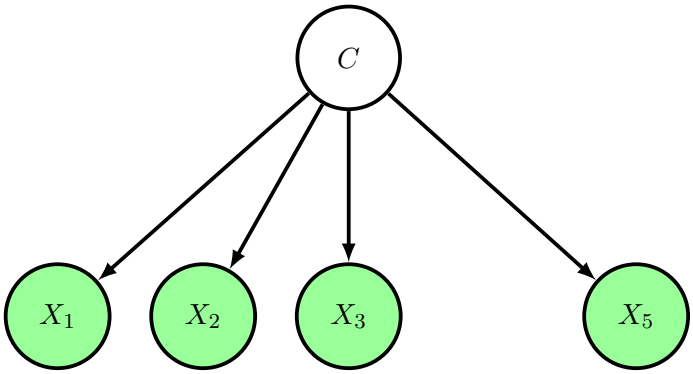

Figure 3.3: Selective naive Bayes classifier structure with four nodes selected from the original set of five nodes, from which $P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*,..,x_5^*) \propto P_C(c)P_{X_1|C}(x_1^*|c)P_{X_2|C}(x_2^*|c)P_{X_3|C}(x_3^*|c)P_{X_5|C}(x_5^*|c)$.

the model has to be built for each feature subset. Simple heuristic methods are used to asses this approach, like greedy search [Langley and Sage, 1994] or floating search [Pernkopf and OLeary, 2003], which is based on a method for adding and on a method for removing attributes and/or arcs from the network structure and capable of removing previously added arcs at a later stage of the search if they turn out to be irrelevant. Nevertheless, owing to the computational cost, heuristics methods are infeasible for a high number of variables. Therefore, combinations of *filter* and *wrapper* approaches are used, creating the *filter-wrapper* method [Inza et al., 2004].

In order to relax the conditional independence assumptions of *naive Bayes* models, it is possible to introduce new features obtained as the Cartesian product of two or more original variables. This is the *semi-naive Bayes* classifier [Pazzani, 1998]. An example of its structure is shown in Fig. (3.4). This model also allows a variable selection. Thus, if L_k with $k = 1,..,T$ is representing the k^{th} feature (original or new feature), Equation (3.6) turns into

$$c^* = \arg\max_c P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*,..x_n^*) = \arg\max_c P_C(c)\prod_{k=1}^{T} P_{X_{L_k}|C}(x_{L_k}^*|c).$$

This model is built from an empty structure and a *forward sequential selection and joining* greedy search [Pazzani, 1998] is used to decide whether (i) add a variable as conditionally independent of the others (original or new variables), or (ii) joining a non-used variable by the current model with each variable (original or new one) already used in the model.

The *tree-augmented naive Bayes* (TAN) classifier [Friedman et al., 1997] keeps the original predictor variables and models the relationships between them, of at most order 1. An example of its structure, which is tree-shaped, is shown in Fig. (3.5). To learn the structure of this classifier, it is necessary to build a directed tree. Kruskal's algorithm [Kruskal, 1956] is used to find the maximum weighted spanning tree. The weight of an edge between X_i and

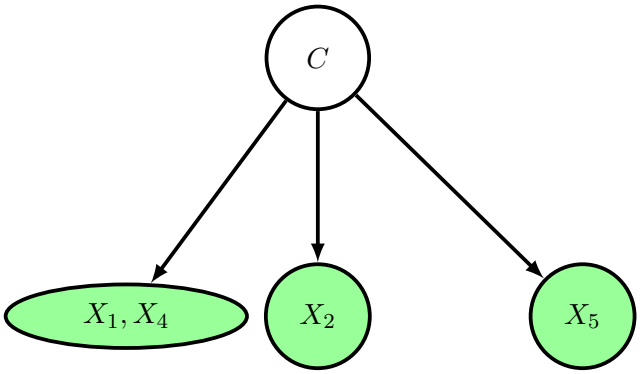

Figure 3.4: Semi-naive Bayes classifier structure with four nodes selected from the original set of five nodes, and two of them joined in a supernode, from which $P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*,..,x_5^*) \propto P_C(c)P_{X_1,X_4|C}(x_1^*,x_4^*|c)P_{X_2|C}(x_2^*|c)P_{X_5|C}(x_5^*|c)$.

X_j is calculated as the conditional MI between the variables given the class C:

$$\mathrm{MI}(X_i,X_j|C) = \int_{\Omega_{X_i}}\int_{\Omega_{X_j}}\sum_c f_{X_i,X_j|C}(x_i,x_j|c)P_C(c)\log\frac{f_{X_i,X_j|C}(x_i,x_j|c)}{f_{X_i|C}(x_i|c)f_{X_j|C}(x_j|c)}dx_idx_j, \tag{3.7}$$

where Ω_{X_i} and Ω_{X_j} represents the domain of variables X_i and X_j respectively, $f_{X_i,X_j|C}(x_i,x_j|c)$ is the joint density function of X_i and X_j given $C=c$, and $f_{X_i|C}(x_i|c)$ and $f_{X_j|C}(x_j|c)$ are the conditional probability density functions of variables X_i and X_j given $C=c$ respectively. Note that if variables X_i and X_j are defined in discrete domains, then the integrals from Equation (3.7) are changed to sums over the values of the variables. This procedure is based on the Chow-Liu algorithm [Chow and Liu, 1968], which approximates a joint probability distribution as a product of second-order conditional and marginal distributions. Thus, this algorithm enables to learn the network structure with no more than second-order relationships. The resulting undirected tree is turned into directed by selecting a random root node and following the unique possible path from that root node, transforming the edges into arcs. For this classification model, Equation (3.6) turns into

$$c^* = \arg\max_c P_{C|\boldsymbol{\mathcal{X}}}(c|x_1^*,..x_n^*) = \arg\max_c P_C(c)P_{X_r|C}(x_r|c)\prod_{i=1,i\neq r}^{n} P_{X_i|C,Pa(X_i)}(x_i^*|c,pa(x_i^*)),$$

where X_r is the selected root node and $Pa(X_i)$ is the only (feature) parent of X_i.

Other Bayesian network classifiers can be found in the literature. There is an extension of the TAN classifier, called *k-dependence Bayesian* classifier [Kohavi, 1996; Sahami, 1996; Zheng and Webb, 2000] that allows more than one predictive variable as parent in the network structure. Bayesian network classifiers that can adopt any Bayesian network structure was studied in Cheng and Greiner [2001]. Furthermore, Bayesian multi-net-based classifiers were proposed by Friedman et al. [1997].

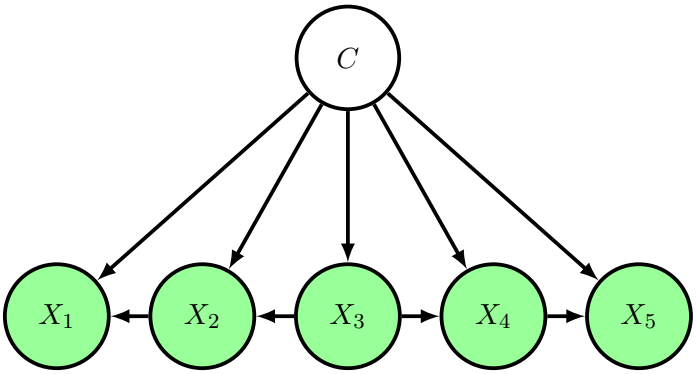

Figure 3.5: Tree-augmented network classifier structure with five nodes, from which $P_{C|\boldsymbol{x}}(c|x_1^*,..,x_5^*) \propto P_C(c)P_{X_1|C,X_2}(x_1^*|c,x_2^*)P_{X_2|C,X_3}(x_2^*|c,x_3^*)P_{X_3|C}(x_3^*|c)P_{X_4|C,X_3}(x_4^*|c,x_3^*)P_{X_5|C,X_4}(x_5^*|c,x_4^*)$ The selected root node of the tree is X_3.

3.4.1.2 Parameter estimation

Estimating the parameters of Bayesian network classifiers can be done in generative or discriminative ways. The generative technique fits the parameters via MLE or Bayesian estimation, both described in Section 3.3.2.2. Discriminative technique consists of finding the parameters that maximize the classification accuracy, which is the main task of Bayesian network classifiers. Alternatively, this technique looks for the maximization of the conditional LL of the class given the predictive variables [Grossman and Domingos, 2004; Greiner et al., 2005; Pernkopf and Bilmes, 2005; Su et al., 2008; Carvalho et al., 2011].

3.5 Software

In this section the tools used in this dissertation for working with Bayesian networks are reviewed. The software used is the **R** software [R Development Core Team, 2008], introduced in Section 2.3.

- The `bnlearn` package [Scutari, 2010] is used for basic manipulation of Bayesian networks structure learning and simple inference with Gaussian networks or discrete networks. This package is available at CRAN repository. It implements for discrete and continuous variables several constraint-based structure learning algorithms, parameter learning, conditional independence tests and network scores. Some Bayesian network classifiers are also implemented, those are the NB and TAN.

- Focused on Bayesian networks classification, the `bnclassify` package [Mihaljevic et al., 2015] provides useful basic algorithms and score functions for discrete variables. It also provide the algorithms for prediction and properties inspection of the implemented classification models, such as the naive Bayes, selective naive Bayes, semi-naive Bayes or tree-augmented naive Bayes. This package is used as a reference to implement several Bayesian network classification models in a different domain.

Chapter 4

Neuroscience

4.1 Introduction

The brain is the most unknown organ from the human body, and its functioning is one of the main challenges of modern sciences. Neuroscience is defined as the science that studies the nervous system. Latest technological and methodological advances have highly improved the insights of this field. However its complexity makes it to remain almost unexplored. The advances in neuroscience are mainly focused on the study of the cerebral cells: the neurons. Thousands of neurons are connected in an extremely complex circuit, with very different activation behaviours and synchronization patterns. Santiago Ramón y Cajal is considered the father of the modern neuroscience for his original investigations of the microscopic structure of the brain. He was awared with the Nobel Prize in Physiology or Medicine in 1906 for his neuroanatomy studies.

Computational sciences and neuroscience converge in the so-called computational neurosciences [Sejnowski et al., 1988; Schwartz, 1993; Dayan and Abbott, 2001; Feng, 2003; Trappenberg, 2009], defined as the theoretical study of the brain used to uncover the principles and mechanisms that guide the development, organization, information-processing and mental abilities of the nervous system.

The applied work developed in this dissertation is focused on the study of neuronal morphology, which analysis and modelling seem to be performed efficiently using machine learning and statistical techniques. This chapter introduces the basic neuroscience notions used in this dissertation.

Chapter outline

Section 4.2 contains some concepts of the organization and structure of the brain, including some basic concepts of the neuronal cells and their classification, given special attention to the pyramidal neuron characteristics as well as their basic structure. In Section 4.3 the nowadays most important modern neuroscience projects are briefly presented.

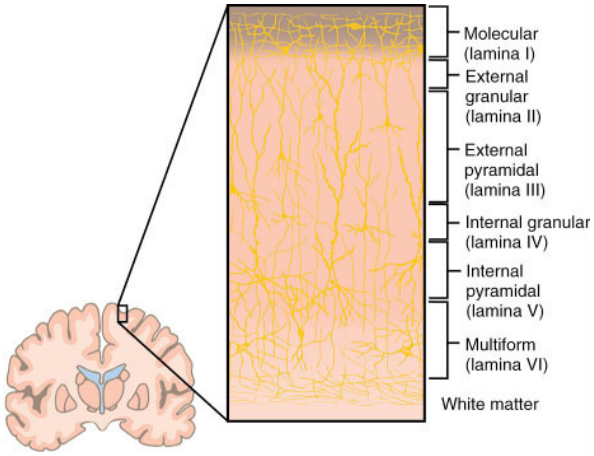

Figure 4.1: Schema of the layers I - VI from the cerebral cortex. Source: Dorland's Medical Dictionary for Health Consumers[1].

4.2 Brain structure

Golgi introduced the concept of the brain as a connected neuronal network [Sporns, 2011]. Santiago Ramón y Cajal followed his research and created the research field known as neuromorphology, the study of the structure of the nervous system.

The brain is an electrical organ. It controls the nervous system, that can be divided into [Kandel et al., 2000]: the medulla oblongata, the pons, the spinal cord, the cerebellum, the midbrain, the diencephalon and the cerebral hemispheres. Each of these has a specific role. Cognitive functions, such as the memory, are located in different areas of the cerebral cortex in the outer part of the cerebral hemispheres.

The cerebral cortex is divided into six layers (Fig. 4.1), named from I (the most superficial layer) to VI (the deepest layer). Each layer has different width. Layer V is sometimes divided into layers Va and Vb, owing to the difference in neuronal density within layer V, where layer Vb has significantly higher neuronal density than layer Va. The cerebral cortex is also divided into cortical areas, depending on their main features. Nevertheless, the number and types of cortical areas are not consolidated, owing to the disagreement between neuroscientists.

Nowadays, the neuromorphology field is divided into two branches: the first is focused on the study of the brain by its synaptic activity (i.e., the signal transmission and reception among neurons), the second is based on the individual analysis of each neuron, finding morphological characteristics that allow the formulation of general neuron structure rules. The neuroscience work presented in this dissertation is based on the latter.

4.2.1 Neurons

Neurons are the most basic working unit of the nervous system. An average human has about 86 billions neurons in his brain [Herculano-Houzel, 2016]. All of them have three basic roles:

[1] https://www.dorlandsonline.com

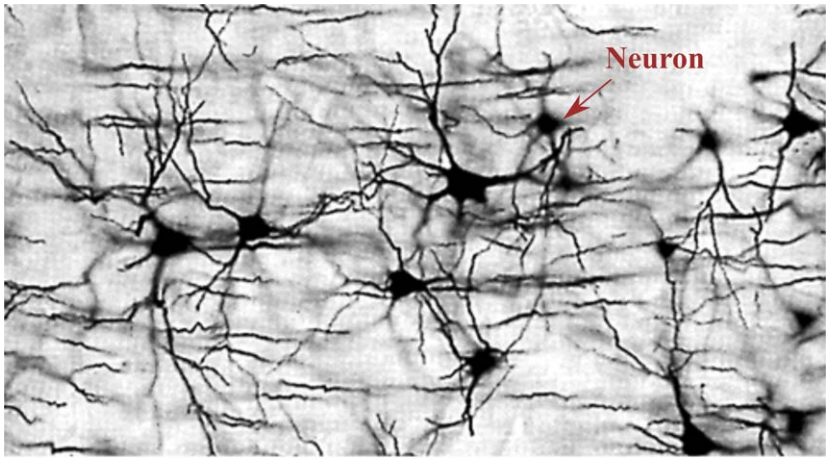

Figure 4.2: Stained neuronal network from Cajal studies. The Golgi method of staining brain tissue renders the neurons and their interconnecting fibres visible in silhouette. Source: Cognitive Consonance[2].

receive signals, integrate them and transmit them.

Following Golgi's research, Santiago Ramón y Cajal used novel staining methods to highlight the parts of the neuronal network (Fig. 4.2). This allowed him to distinguish that each neuron is a single cell separated from the others. This discovery is known as the neuron doctrine. Furthermore, he found that neurons communicate among them by sending electrical signals through a prolongation of the neuron called axon. These signals are received through other prolongations of the neuron, called dendrites.

4.2.1.1 Structure

Neurons are composed of three main parts (Fig. 4.3): dendrites, cell body (called soma) and axon. Most neurons follow the same general structural plan. Nevertheless the structure of individual neurons may vary depending on their specific function [Jacobs et al., 2001; Elston et al., 2005; Benavides-Piccione et al., 2006; Komendantov and Ascoli, 2009].

- The soma includes the cellular structures such as the nucleus, mitochondria, etc. The dendrites and the axon grow from the soma, which morphology is highly affected by the number of dendrites and the orientation of them together with the axon. The chemical processing of incoming information takes place mainly in the soma. There is not an established solid definition for the soma boundaries. Therefore, usually experts must trace these boundaries under their criteria.

- The dendrites are composed by the dendritic arbors and the dendritic spines. The dendritic arbors are prolongations that born in the soma and are subsequently bifurcated creating branched structures. These are covered by thousands of small membranous

[2]http://cognitiveconsonance.info/

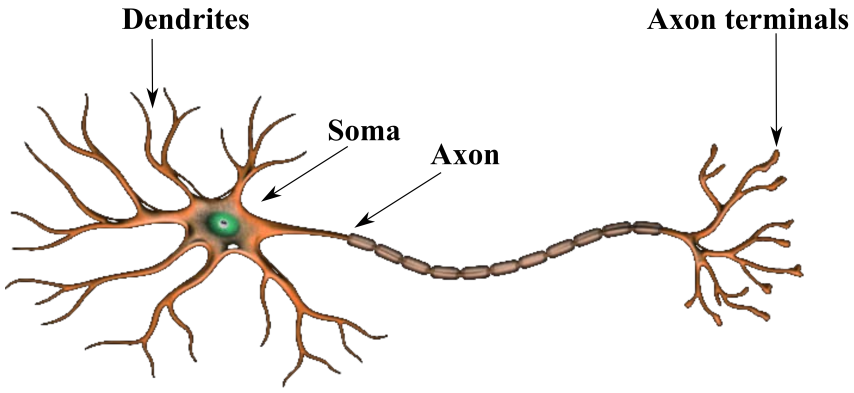

Figure 4.3: Basic structure of a neuron.

bumps, called dendritic spines, each of which takes part of the neuron-to-neuron connection, called synapses [Nimchinsky et al., 2002]. Most neurons receive the signals through their dendritic spines.

Some neurons show two types of dendrites: basal dendrites, which arise from the soma creating a spherical arborization, and the apical dendrite, that arises from the apex of pyramidal soma .

- The axon is also a prolongation of the neuron. It is born in the soma, it can be extended a long distance from it and its diameter is smaller than the dendrites diameter. Usually, the axon grows until it connects to a group of dendrites of other neurons (i.e., the axon terminals connect with the dendritic spines of other neurons). The function of the axon is to transmit the information from the axon terminals to the dendritic spines through synapses. Some neurons are axonless [Wu et al., 2011], nevertheless they are not covered in this dissertation.

4.2.1.2 Classification

Based on their functions, neurons can be divided into three main classes: sensory neurons, motor neurons and interneurons (Fig. 4.4).

The function of sensory neurons is to get information about what happen with anything related to the body, either inside or outside it, e.g., when somebody is cooking and touches the pan, the sensory neurons transmit the information that the pan is hot to the brain.

Motor neurons are those that are responsible of getting information from the sensory neurons to transmit orders to the muscles, organs or glands, e.g., following the sensory neurons example, when they transmit the information about the hot pan, the motor neurons will order the body part that was in contact with the pan to come off it.

The interneurons have the function to connect one neuron to another (i.e., receptors from either sensory neurons or interneurons and transmitters to either motor neurons or interneuros). Despite there is not an universal accepted catalogue of interneurons, they

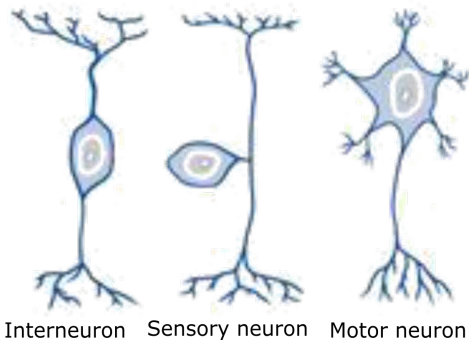

Figure 4.4: Different types of neurons based on their functions. Source: Wichterle et al. [2013].

can be also subclassified depending on their morphological characteristics [DeFelipe, 1993; Mihaljević et al., 2014, 2015b,a].

Different types of neurons show great diversity in size and shape (Fig. 4.5) [P.I.N.G, 2008; Ding and Glanzman, 2011]. Perhaps, the best-known of these types of neurons are the pyramidal neurons, which are the most abundant type of neuron in the cerebral cortex (about 70-85%).

4.2.1.3 Pyramidal neurons

Pyramidal neurons were discovered by Santiago Ramón y Cajal. Their name comes from their pyramidal shape soma. These neurons are characterized by their large apical dendrites and short basal dendrites (Fig. 4.6), which represent about the 90% of the dendritic length of cortical pyramidal neurons from layers II, III and V [Larkman, 1991]. They can be found in many different areas of the brain such as the cortex and hippocampus, and their characteristics may vary depending on their location. Furthermore, the pyramidal neurons soma is usually represented as a tetrahedron with an acute angle pointing towards the cerebral cortex surface.

Neurons in the cortex can be excitatory, which release the neurotransmitter glutamate, and inhibitory, which release γ-amino-butyric acid (GABA). Pyramidal neurons are the most abundant excitatory neurons [Spruston, 2008]. They are sometimes enwrapped by the axons of inhibitory basket cells, which are interneurons that control and refine the firing of pyramidal neurons through their inputs [David and Pierre, 2009].

The structure of pyramidal neurons may vary between regions or layers (Fig. 4.7). Nevertheless, all of them share the separation between apical and basal dendrites characterization [Spruston, 2008]. Furthermore, both basal and apical dendrites are studded with dendritic spines.

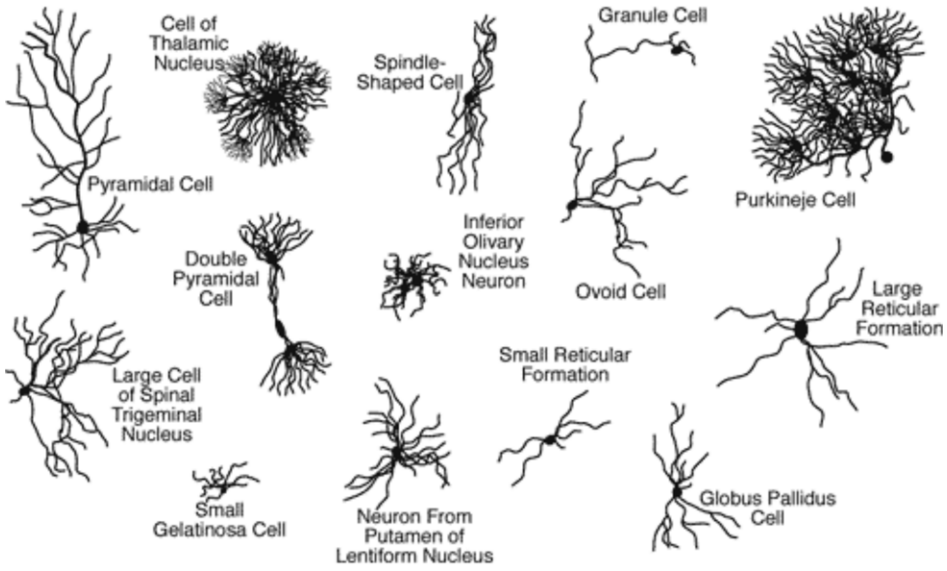

Figure 4.5: Different types of neurons by shapes and sizes based on the drawings made by Ramón y Cajal. The types of neurons shown are Pyramidal, double Pyramidal, Thalamic nucleus, Spinal Trigeminal nucleus, inferior Olivary nucleus, Putamen of Lentiform nucleus, Spindle-shaped, Granule, Purkinje, Ovoid, large Reticular, small Reticular, small Gelatinosa and Globus Pallidus. Source: The Mind Project [Stufflebeam, 2008] .

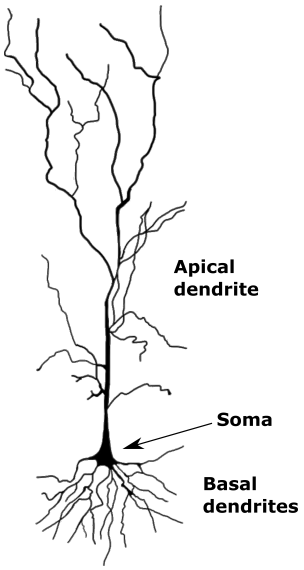

Figure 4.6: Schema of the morphology of a typical pyramidal neuron. This neuron has been obtained from layer V in the rat somatosensory cortex.

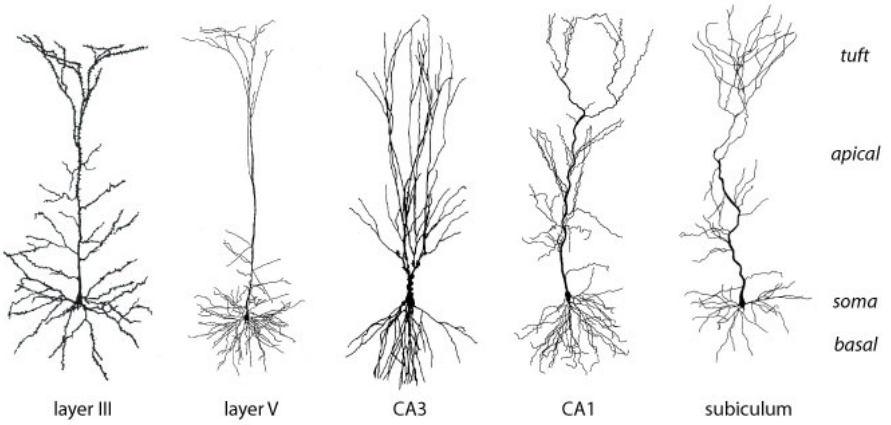

Figure 4.7: A variety of pyramidal neurons from different parts of the brain. CA3 and CA1 are the Cornu ammonis areas 3 and 1 respectively, located next to the subiculum, all of them from the Hippocampus of the cerebral cortex. Source: Spruston [2008].

4.3 Current neuroscience research projects

Recently, the researching community has shown an important interest in the study of the brain. The Blue Brain Project[3] (BBP) [Markram, 2006] is an ambitious project from the Brain Mind Institute at the École Polytechnique Fédérale de Lausanne (EPFL) and International Business Machines (IBM). The goal of the BBP is to build biologically detailed digital reconstructions and simulations of the rodent, and ultimately the human brain. It started on 2005, headed by Henry Markram from the EPFL (Switzerland) under the premise of assimilating the wealth of data that have been produced for neuroscience studies over the past century in order to build accurate models of the brain. For this purpose, the project counts with the IBM's Blue Gene supercomputer, a massively parallel, tightly interconnect machine with 65.536 processors, 839 Teraflops of peak performance, 65 TeraBytes of RAM, 128 TeraBytes of BlueGene Active Storage (BIGAS) and more than 4 PetaBytes of hard disk that provide a high level of detail at which the brain can be modelled.

In 2006, the BBP announced the first rat's neocortical column model, that was the initial objective of the project. Its website states that the results produced by this project will provide the capability to model and simulate: the brain or any region of the brain of any species, at any stage in its development; specific pathologies of the brain; and diagnostic tools and treatments for these pathologies. The geometric and computational models of the brain produced by the facility will reproduce the structural and functional features of the biological brain with electron microscopic and molecular dynamic level accuracy.

Spanish representations, through the Universidad Politécnica de Madrid (UPM) and Instituto Cajal (IC) from Consejo Superior de Investigaciones Científicas (CSIC), are involved in the BBP with an initiative named Cajal Blue Brain Project[4] (CBBP), presented at the

[3] http://bluebrain.epfl.ch/
[4] http://cajalbbp.cesvima.upm.es/

beginning of 2009. Different research groups and laboratories from Spanish institutions take part in this initiative, grouping together a large number of scientists, engineers and practitioners. This interdisciplinary project requires the expert-knowledge of scientists from various researching fields. As stated on its website, the aim of the CBBP is to achieve the following objectives:

- To decode the synaptome or detailed map of the synaptic connections of the cortical column and, as a result, reconstruct all its components;
- To give a strong boost to research on the cortical column, exploring in depth current hypotheses about its normal function and dysfunctions (especially Alzheimer rat's disease);
- To devise new methods to process and analyse the experimental data obtained in the aforementioned research studies;
- To develop computer technology to study neuronal functions using graphical tools and visualization methods.

And secondarily:

- To understand the implication of glial cells and blood vessels in the organization of the cortical column;
- To study the modulation of the functional organization of the cerebral cortex by cortical and subcortical afferent connections;
- To decipher the functional organization of cortical circuits in vitro;
- To simulate in silico the activity of the cortical column by means of a supercomputer.

In 2013, two remarkable projects focused on the study of the brain were presented: the Human Brain Project (HBP) in Europe and the Brain Research through Advancing Innovative Neurotechnologies (BRAIN) in the United States.

The HPB[5] [Markram, 2012] is one of the two selected projects for the European Commission's Future and Emerging Technologies (FET) Flagship Program, which strives to accelerate the fields of neuroscience, computing and brain-related medicine. This project has the following main objectives:

- To create and operate a European Scientific Research Infrastructure for brain research, cognitive, neuroscience, and other brain-inspired sciences
- To gather, organise and disseminate data describing the brain and its diseases;
- To simulate the brain;

[5]HBP url: http://humanbrainproject.eu/

- To build multi-scale scaffold theory and models for the brain;
- To develop brain-inspired computing, data analytics and robotics;
- To ensure that the HBP's work is undertaken responsibly and that it benefits society.

The BRAIN[6] [Alivisatos et al., 2012, 2013], also called Brain Activity Map (BAM) project is the United States counterpart of the HPB, but completely independent and presented by former President Obama almost at the same time. This project is also aimed to revolutionizing our understanding of the human brain. In [Alivisatos et al., 2013] the main objectives of the project are presented. These are:

- To build new classes of tools that can simultaneously image or record the individual activity of most, or even all, neurons in a brain circuit, including those containing millions of neurons;
- To create tools to control the activity of every neuron individually in these circuits, because testing function requires intervention;
- To understand circuits' function.

The successful development of these projects depends on the strides in statistics and computer sciences, as well as the capacity to gather the required data. In particular, the input of data must be done, as well as the programming of models and simulations in order to find those biological keys for breakthrough discovers in the study of the brain. Thus, these are multidisciplinary projects that serve as an example of how the connection between several different disciplines may lead to important insights for the human being evolution.

[6]BRAIN url: http://braininitiative.nih.gov/

Part III

CONTRIBUTIONS TO BAYESIAN NETWORKS AND DIRECTIONAL STATISTICS

Chapter 5

Circular Bayesian classifiers using wrapped Cauchy distributions

5.1 Introduction

As stated in Chapter 2, the natural periodicity of circular data sometimes makes traditional statistics methods ineffective. In addition, in Chapter 3 we reviewed probabilistic graphical models [Koller and Friedman, 2009]. We outlined some of the advantages of using probabilistic graphical models, such as the fact that they are easily interpreted, they handle missing data effectively and they can cope with inference and learning tasks. In particular we explained that Bayesian networks [Pearl, 1988] can deal efficiently with supervised classification (i.e., via the Bayesian network classifiers [Bielza and Larrañaga, 2014]) and offer an explicit, graphical and interpretable representation of uncertain knowledge, which has made it possible to successfully apply them to real-world problems.

Yet circular data has been commonly treated as linear data in supervised classification tasks. Only a few circular classifiers exist, and almost none of them are based on the principles of Bayesian networks, capable of capturing multivariate relationships among variables. Most of them focus on discriminant analysis and assume several circular distributions such as the vM distribution [Morris and Laycock, 1974], later extended to the von Mises-Fisher distribution [Romanazzi, 2014]. SenGupta and Roy [2005] used a classification discriminant rule based on the mean chord-length to classify a new observation into one of two different circular populations that are vM, when training samples are available for each of them. Also a likelihood ratio test based on a bootstrapping approach for classifying into two populations was proposed for linear and circular data [SenGupta and Ugwuowo, 2011]. Kirby and Miranda [1996] proposed a variation of an artificial neural network, including a circular node, which was able to keep and send circular information. More recently, Fernandes and Cardoso [2016] proposed a binary circular logistic regression as the discriminative counterpart to the naive Bayes model, which does not make assumptions on the input data distribution. López-Cruz et al. [2015] is the only study in which Bayesian classifiers were used. For the vM and vM-

Fisher distributions, López-Cruz et al. proposed adaptations of the naive Bayes classifier and selective naive Bayes classifier, which are two of the simplest and best-known supervised classification models based on Bayesian network principles.

The lack of Bayesian network classifiers for circular data is due to the absence of circular Bayesian network models, which are very difficult to develop because of their circular multivariate distribution nature. A family of distributions is said to be closed under marginalization and conditioning when the marginals and conditionals of the multivariate distribution follow the same distribution. However, the marginals and conditionals of most circular distributions do not belong to the same family of distributions, making the modelling phase and posterior inference processes difficult.

We presented the vM distribution in Section 2.2.3.1. A bivariate vM distribution also exists and was introduced by Mardia [1975], who subsequently extended it to the multivariate case [Mardia et al., 2008]. He showed that the conditional distributions are also vM distributions. Nevertheless, the marginal distributions are either unimodal or bimodal, and only the unimodal case could be approximated to a vM distribution when the concentration parameter is large. Therefore, as explained in [Bielza and Larrañaga, 2014] for discrete distributions, it would be much more complicated to achieve an efficient learning and inference. Therefore, we ruled out the use of vM distributions for our particular purpose. In Section 2.2.3.2 we overviewed the wC distribution. Kato and Pewsey [2015] developed a five-parameter bivariate wC distribution for toroidal data, whose marginals and conditionals follow univariate wC distributions. This family of bivariate wC distributions is therefore closed under conditioning and marginalization. Following these properties, we develop the first tree-structured Bayesian network model that deals with circular data which follows a wC distribution [Leguey et al., 2016a]. However, this model only accounts for the discovery of conditional independence relationships of a set of random variables, without considering any as a class variable. This is a specificity of supervised classification problems that requires special learning algorithms.

In this chapter, building on previous work regarding supervised classification using Bayesian networks for circular statistics, we propose four circular Bayesian network classification models capable of dealing with supervised data following wC distributions. The models to be presented are called wrapped Cauchy naive Bayes (wCNB), wrapped Cauchy selective naive Bayes (wCsNB), wrapped Cauchy semi-naive Bayes (wCsmNB) and wrapped Cauchy tree-augmented naive Bayes (wCTAN) classifiers.

Chapter outline

Section 5.2 reviews the bivariate wC distribution of Kato and Pewsey [Kato and Pewsey, 2015]. Section 5.3 describes the four wC classifiers presented in this Chapter. In Section 5.4, we assess the four models in synthetic domains, requiring the design of a simulation method for these wC Bayesian network classifiers. Finally, Section 5.5 provides concluding remarks and proposals for future work.

5.2 Wrapped Cauchy distribution

5.2.1 Definitions

A five-parameter bivariate wC distribution was proposed by Kato and Pewsey [Kato and Pewsey, 2015]. A random vector (Θ_1,Θ_2) that follows the five-parameter circular bivariate wC distribution [Kato and Pewsey, 2015], denoted by $bwC(\mu_1,\mu_2,\varepsilon_1,\varepsilon_2,\rho)$, has the density function

$$\begin{aligned} f(\theta_1,\theta_2) = c[&c_0 - c_1\cos(\theta_1-\mu_1) - c_2\cos(\theta_2-\mu_2) - c_3\cos(\theta_1-\mu_1)\cos(\theta_2-\mu_2) \\ &-c_4\sin(\theta_1-\mu_1)\sin(\theta_2-\mu_2)]^{-1}, \quad \text{with} \quad \theta_1,\theta_2\in(-\pi,\pi], \end{aligned} \tag{5.1}$$

where $c = (1-\rho^2)(1-\varepsilon_1^2)(1-\varepsilon_2^2)/4\pi^2$, $c_0 = (1+\rho^2)(1+\varepsilon_1^2)(1+\varepsilon_2^2) - 8|\rho|\varepsilon_1\varepsilon_2$, $c_1 = 2(1+\rho^2)\varepsilon_1(1+\varepsilon_2^2) - 4|\rho|(1+\varepsilon_1^2)\varepsilon_2$, $c_2 = 2(1+\rho^2)(1+\varepsilon_1^2)\varepsilon_2 - 4|\rho|\varepsilon_1(1+\varepsilon_2^2)$, $c_3 = -4(1+\rho^2)\varepsilon_1\varepsilon_2 + 2|\rho|(1+\varepsilon_1^2)(1+\varepsilon_2^2)$, $c_4 = 2\rho(1-\varepsilon_1^2)(1-\varepsilon_2^2)$, $\mu_1,\mu_2\in(-\pi,\pi]$, $\varepsilon_1,\varepsilon_2\in[0,1)$, and $\rho\in(-1,1)$. Here, μ_1 and μ_2 are the marginal location parameters, ε_1 and ε_2 are the marginal concentration parameters, and ρ is the parameter controlling the association between Θ_1 and Θ_2, from total independence ($\rho=0$) to perfect correlation ($\rho=\pm1$). When $\varepsilon_1>0$ and $\varepsilon_2>0$, f in Equation (5.1) is unimodal and pointwise symmetric about (μ_1,μ_2).

Using the complex form to represent univariate and bivariate wC models can simplify their computation issues [McCullagh, 1996; Kato and Pewsey, 2015].

Let $Z = e^{(i\Theta)}$, where Θ is distributed as the univariate wC given by Equation (2.4). Then, Z has a density function given by

$$f(z) = \frac{1}{2\pi}\frac{|1-|\lambda|^2|}{|z-\lambda|^2}, \quad z\in\Omega,\ \lambda\in\hat{\mathbb{C}}\setminus\Omega, \tag{5.2}$$

where $\lambda = \varepsilon e^{(i\mu)}$, $\hat{\mathbb{C}} = \mathbb{C}\cup\{\infty\}$, and $\Omega = \{z\in\mathbb{C}:|z|=1\}$. Similarly to [McCullagh, 1996], we denote Z distributed as in Equation (5.2) as $Z\sim C^*(\lambda)$.

Let $(Z_1,Z_2) = (e^{(i\Theta_1)}, e^{(i\Theta_2)})$, where (Θ_1,Θ_2) is distributed as the bivariate wC in Equation (5.1). Then, the density function of (Z_1,Z_2) is

$$f(z_1,z_2) = \frac{(4\pi^2)^{-1}(1-\rho^2)(1-\varepsilon_1^2)(1-\varepsilon_2^2)}{|a_{11}(\overline{z}_1\eta_1)^q z_2\overline{\eta}_2 + a_{12}(\overline{z}_1\eta_1)^q + a_{21}z_2\overline{\eta}_2 + a_{22}|^2}, \quad z_1,z_2\in\Omega, \tag{5.3}$$

where $q\in\{-1,1\}$ is the sign of ρ, $\eta_k = e^{(i\mu_k)}\in\Omega$ with $k\in\{1,2\}$, $\overline{z}_k$ is the complex conjugate of z_k, $\varepsilon_1,\varepsilon_2\in[0,1)$, $\rho\in(-1,1)$, and

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} \varepsilon_1\varepsilon_2 - |\rho| & |\rho|\varepsilon_2 - \varepsilon_1 \\ |\rho|\varepsilon_1 - \varepsilon_2 & 1 - |\rho|\varepsilon_1\varepsilon_2 \end{pmatrix}. \tag{5.4}$$

We denote (Z_1,Z_2) distributed as in Equation (5.3) as $(Z_1,Z_2)\sim bC^*(\eta_1,\eta_2,\varepsilon_1,\varepsilon_2,\rho)$.

This five-parameter bivariate wC complex form representation verifies the following result:

Theorem 5.1. (Kato and Pewsey [Kato and Pewsey, 2015]) *A random vector* (Z_1, Z_2) *with density as in Equation (5.3) has marginals* $Z_1 \sim C^*(\varepsilon_1\eta_1)$ *and* $Z_2 \sim C^*(\varepsilon_2\eta_2)$*, and conditionals* $Z_1|Z_2 = z_2 \sim C^*(-\eta_1[\boldsymbol{A} \circ (z_2\overline{\eta}_2)^q])$ *and* $Z_2|Z_1 = z_1 \sim C^*(-\eta_2[\boldsymbol{A}^T \circ (z_1\overline{\eta}_1)^q])$*, where* $\boldsymbol{A}$ *is defined as in Equation (5.4),* $\boldsymbol{A}^T$ *is the transpose of* $\boldsymbol{A}$*, and*

$$\boldsymbol{A} \circ z = \frac{a_{11}z + a_{12}}{a_{21}z + a_{22}}.$$

As far as we know, there is no other bivariate circular distribution for which conditional and marginal distributions belong to the same family. Therefore, we consider the wC distribution to be suitable for developing our classification models, as the requirements for the classifier structures that we will develop are of at most a tree-structure (i.e., only bivariate, marginal and conditional densities are required). Furthermore, we require the definition of a conditional circular mutual information measure between variables that follow wC distributions. Hence, the development of these classification models and their corresponding learning algorithms is suitable and far from straightforward.

5.2.2 Parameter estimation

Working with the density given by Equation (5.1), numerical methods have to be used to find the parameter estimates, since there is no closed-form expression for the maximum likelihood estimates. Kato and Pewsey [Kato and Pewsey, 2015] demonstrated that the method of moments [Bowman and Shenton, 1985] is more efficient (see Section 2.2.3.2); it is computationally very fast, easy to implement and with closed form formulas for the parameter estimates.

Let $\{(\theta_{1j}, \theta_{2j}) \ , j = 1, ..., N\}$ be a random sample from a $bwC(\mu_1, \mu_2, \varepsilon_1, \varepsilon_2, \rho)$ (Equation (5.1)). Then, the estimators obtained using the method of moments for $\mu_1, \mu_2, \varepsilon_1, \varepsilon_2$ and ρ are [Kato and Pewsey, 2015]

$$\hat{\mu}_k = arg(\bar{R}_k), \quad \hat{\varepsilon}_k = |\bar{R}_k|, \quad k = 1, 2,$$

with $\overline{R}_k = \frac{1}{N}\sum_{j=1}^{N} e^{\left(i\theta_{kj}\right)}$, and

$$\hat{\rho} = \frac{1}{N}\left(|\sum_{j=1}^{N} e^{(i(\Phi_{1j} - \Phi_{2j}))}| - |\sum_{j=1}^{N} e^{(i(\Phi_{1j} + \Phi_{2j}))}|\right), \tag{5.5}$$

where $\Phi_{rj} = 2\arctan\left(\frac{1+\hat{\varepsilon}_r}{1-\hat{\varepsilon}_r}\tan\left(\frac{\theta_{rj} - \hat{\mu}_r}{2}\right)\right), \quad r = 1, 2.$

5.3 Wrapped Cauchy classifiers

Let $\boldsymbol{\Theta} = (\Theta_1, .., \Theta_n)$ be a vector of circular predictor random variables or features, and let C be a discrete class variable which takes values (labels) in the set $\Lambda(C)$. Given a sample of N labelled instances $(\boldsymbol{\Theta}^1, C^1), .., (\boldsymbol{\Theta}^N, C^N)$, the supervised classification problem consists in

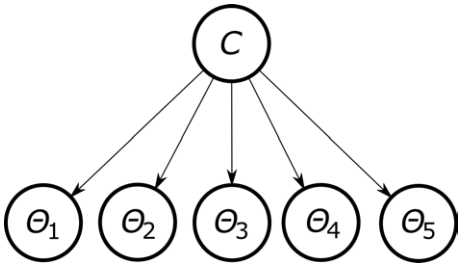

Figure 5.1: wCNB structure with five circular predictor nodes, from which $p(c|\boldsymbol{\theta}) \propto p(c) f_{\Theta_1|c}(\theta_1|c) f_{\Theta_2|c}(\theta_2|c) f_{\Theta_3|c}(\theta_3|c) f_{\Theta_4|c}(\theta_4|c) f_{\Theta_5|c}(\theta_5|c)$.

developing a model capable of assigning a class label to a new object based on the values of its features.

In Section 3.4 we overview the Bayesian network classifiers [Bielza and Larrañaga, 2014]. Our purpose is to develop the circular domain counterpart of the well-known Bayesian network classifiers described in Section 3.4.1.1 (naive Bayes, selective naive Bayes, semi-naive Bayes and tree-augmented naive Bayes), when the underlying variables follow wC distributions.

5.3.1 Wrapped Cauchy naive Bayes

The wrapped Cauchy naive Bayes (wCNB) classifier is the simplest of the four Bayesian network classifier models that we present in this Chapter, where C is the parent of all circular features and these are assumed to be conditionally independent among them given C (Fig. 5.1)

$$p(C = c|\boldsymbol{\Theta} = \boldsymbol{\theta}) \propto p(C = c) \prod_{i=1}^{n} f_{\Theta_i|C=c}(\theta_i|c). \tag{5.6}$$

The wCNB determines the class value c^* for a new instance using a maximum a posteriori decision rule

$$c^* = \arg \max_{c \in \Lambda(C)} p(C = c|\boldsymbol{\Theta} = \boldsymbol{\theta}).$$

Since each predictor variable Θ_i given $C = c$ follows a wC distribution with location parameter $\mu_{i,c}$ and concentration parameter $\varepsilon_{i,c}$, we can express Equation (5.6) as

$$p(c|\boldsymbol{\theta}) \propto= \frac{p(C = c) \prod_{i=1}^{n} \alpha_{i,c}}{\prod_{i=1}^{n} (1 - \beta_{i,c})}, \tag{5.7}$$

where $\alpha_{i,c} = \frac{1}{2\pi} \frac{(1-\varepsilon_{i,c}^2)}{(1+\varepsilon_{i,c}^2)}$ and $\beta_{i,c} = \frac{2\varepsilon_{i,c} \cos(\theta_i - \mu_{i,c})}{(1+\varepsilon_{i,c}^2)}$.

5.3.2 Wrapped Cauchy selective naive Bayes

As explained in Section 3.4.1.1, sometimes there are several predictor variables that do not contribute to classification (i.e., they are redundant), and NB classifier is affected by such variables [Langley and Sage, 1994]. Therefore, FSS techniques [Saeys et al., 2007] could increase the accuracy of the classification model significantly [Blanco et al., 2005]. Wrapped

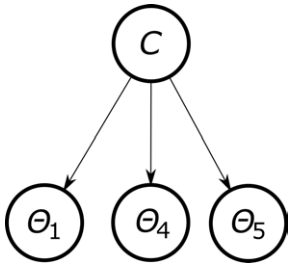

Figure 5.2: wCsNB structure with three nodes selected from the original set of five predictive variables, from which $p(c|\boldsymbol{\theta}) \propto p(c) f_{\Theta_1|c}(\theta_1|c) f_{\Theta_4|c}(\theta_4|c) f_{\Theta_5|c}(\theta_5|c)$.

Cauchy selective naive Bayes (wCsNB) is a classification model with a structure similar to that of wCNB, but not all the variables are necessarily used by the classifier. FSS techniques were previously employed in a circular classification model with vM and vM-Fisher distributions in [López-Cruz et al., 2015], where a filter-wrapper algorithm is applied to rank the variables according to the MI between them and the class, and therefore, using the ranking provided by the filter step, the variables are selected to induce a new classifier until the best model is achieved.

We also use a filter-wrapper algorithm (see Section 3.4.1.1) at this point. The filter step is based on the computation of the MI between each circular variable and the class, followed by the ranking of the predictive variables according to their MI value. Since there is no equation to compute the MI between circular and discrete variables, we approach the problem using Monte Carlo methods, as in [López-Cruz et al., 2015]; we model the conditional density functions of $\Theta_i|C = c$ as wC distributions. Hence

$$\mathrm{MI}(\Theta_i, C) \approx \frac{1}{M} \sum_{j=1}^{M} \log \frac{\hat{f}_{\Theta_i|c^{*(j)}} \left(\theta_i^{*(j)}|c^{*(j)}\right) \hat{p}\left(C = c^{*(j)}\right)}{\hat{f}_{\Theta_i}\left(\theta_i^{*(j)}\right) \hat{p}\left(C = c^{*(j)}\right)}, \tag{5.8}$$

where M is the number of instances $\left(\theta_i^{*(j)}, c^{*(j)}\right)$ sampled from $\hat{f}_{\Theta_i|c}(\theta_i|c)\,\hat{p}(C = c)$, with $\hat{f}_{\Theta_i|c}(\theta_i|c)$ the fitted wC density function of the conditional density function of Θ_i given $C = c$, and $\hat{p}(C = c)$ the relative frequency of instances that belong to class c in the training set.

The wrapper step consists in creating a new classifier by deciding whether or not to include the predictive variables from the ranked list of the filter step. Each iteration of the wrapper step induces a new classifier adding the next predictive variable from the list. This step finishes when no accuracy improvement is achieved by including the next predictive variable from the ranked list. This model is similar to the wCNB, but including only the selected wC variables (set S) (Fig. 5.2) and therefore

$$p(c|\boldsymbol{\theta}) \propto p(c|\boldsymbol{\theta}_S) = p(C = c) \prod_{i \in S} f_{\Theta_i|C=c}(\theta_i|c). \tag{5.9}$$

As for the wCNB, the wCsNB determines the class value c^* for a new instance using a maximum a posteriori decision rule

$$c^* = \arg\max_{c \in \Lambda(C)} p(C = c | \boldsymbol{\Theta_S} = \boldsymbol{\theta_S}).$$

Likewise for Equation (5.7), we can express Equation (5.9) as

$$p(c|\boldsymbol{\theta_S}) = \frac{p(C = c) \prod_{i \in S} \alpha_{i,c}}{\prod_{i \in S}(1 - \beta_{i,c})},$$

where $\alpha_{i,c} = \frac{1}{2\pi} \frac{(1-\varepsilon_{i,c}^2)}{(1+\varepsilon_{i,c}^2)}$ and $\beta_{i,c} = \frac{2\varepsilon_{i,c} \cos(\theta_{i,c} - \mu_{i,c})}{(1+\varepsilon_{i,c}^2)}$.

5.3.3 Wrapped Cauchy semi-naive Bayes

Usually, the assumption of conditional independence between predictive variables given the class variable is dismissed. In Section 3.4.1.1 we state that the semi-naive Bayes classifier considers dependencies between predictive variables.

Our proposal for this model, called wrapped Cauchy semi-naive Bayes (wCsmNB) classifier, takes into account the possible dependence between predictive wC variables by introducing new features obtained as the Cartesian product of two of the original circular predictor variables. Thus we work with a bivariate wC distribution. These new features remains conditionally independent given the class variable.

Given L_k with $k = 1, ..., T$, representing the kth feature (original or new features)

$$p(c|\boldsymbol{\theta}) \propto p(C = c) \prod_{k=1}^{T} f_{\Theta_{L_k}|C=c}(\boldsymbol{\theta}_{L_k}|c).$$

To determine those original variables that are candidates to create new features from the Cartesian product between them, we develop an adaptation of the *forward sequential selection and joining* (FSSJ) algorithm [Pazzani, 1998] described in Algorithm 5.1. It is important to note that once the new features are created by joining two original features, these new features cannot be used to create others. However, these new features can be separated in order to use one of the two original features to create another new feature by joining with a different original feature that had not yet been added to the model. This algorithm may result in a selection of variables that provide the best achievable solution, before all of the original variables are included in the model (Fig. 5.3).

Again, as for the previous models presented in this section, the wCmNB determines the class value c^* for a new instance using a maximum a posteriori decision rule

$$c^* = \arg\max_{c \in \Lambda(C)} p(C = c | \boldsymbol{\Theta} = \theta_{L_k}).$$

Algorithm 5.1 Adaptation of the FSSJ algorithm of [Pazzani, 1998]

1: Let T be the variable list, initialized as $T = \emptyset$.
2: Given $\Theta_1, \Theta_2, ..., \Theta_n$ circular wC predictor variables from a variable list A, move the first variable from A to T.
3: Move the next variable from A to T, considering:
   - Joining the variable to another variable currently in T. If the latter variable was previously joined to another variable from T, remove this from T and add it to A, and consider adding it later.
   - Add the variable as conditionally independent of the other variables given C to the current classifier.
4: Repeat Step 3 until the best model is achieved

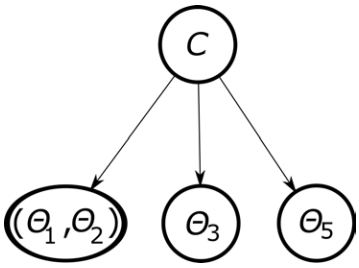

Figure 5.3: wCsmNB structure with four nodes from the original set of five predictive variables, from which $p(c|\boldsymbol{\theta}) \propto p(c) f_{\Theta_1,\Theta_2|c}(\theta_1, \theta_2|c) f_{\Theta_3|c}(\theta_3|c) f_{\Theta_5|c}(\theta_5|c)$.

5.3.4 Wrapped Cauchy tree-augmented naive Bayes

Wrapped Cauchy tree-augmented naive Bayes (wCTAN) classifier is a variation of the TAN classifier (Section 3.4.1.1) with the novelty of the allowance of the use of wC circular variables for predictive features. wCTAN assumes that the class variable has no parents, and the rest of the variables have at most one other variable as parent apart from C (Fig. 5.4).

The process for building a wCTAN is summarized in the following three steps:

- Step 1: The structure of the tree for predictive features is learned using Algorithm 5.2. We use the conditional circular mutual information, denoted as $\mathrm{MIC}(\Theta_i, \Theta_j|C)$, which is defined as

$$\mathrm{MIC}(\Theta_i, \Theta_j|C) = \sum_c \mathrm{MIC}(\Theta_i, \Theta_j|C = c) p(C = c),$$

  with

$$\mathrm{MIC}(\Theta_i, \Theta_j|C = c) = \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} f(\theta_i, \theta_j|c) \log\left(\frac{f(\theta_i, \theta_j|c)}{f(\theta_i|c) f(\theta_j|c)}\right) d\theta_i d\theta_j.$$

  where the marginal density functions given the class, $f(\theta_i|c)$ and $f(\theta_j|c)$, and the joint density function given the class, $f(\theta_i, \theta_j|c)$, have been previously estimated from data. This structure learning algorithm (Algorithm 5.2) is based on score and search, where structure learning is posed as an optimization problem, using a maximum weighted

spanning tree algorithm (where the weights are given by the MIC), a variant of the Chow Liu algorithm [Chow and Liu, 1968].

Algorithm 5.2 Adaptation of the Chow Liu algorithm of [Chow and Liu, 1968]

1: Given $\Theta_1, \Theta_2, ..., \Theta_n$ wC variables, estimate the bivariate joint density function $f(\theta_i, \theta_j|c)$ for all pairs of variables, and the marginals $f(\theta_i|c)$, for each $c \in \Lambda(C)$, $i, j = 1, ..., n$
2: Using these, compute all conditional MIC$(\Theta_i, \Theta_j|C)$ values, (i.e., the $n(n-1)/2$ edge weights) and order them
3: Assign the largest two edges to the undirected tree to be represented
4: Examine the next-largest edge, and add it to the tree unless it forms a loop, in which case discard it and examine the next largest edge
5: Repeat Step 4 until $n-1$ edges have been selected (and the spanning undirected tree is finished)

For Step 1 in Algorithm 5.2, the estimate of the bivariate and marginal densities are performed for each c using the methods explained in Section 5.2. Like the traditional mutual information measure for linear variables, the MIC$(\Theta_i, \Theta_j|C)$ denotes the entropy reduction of Θ_i (Θ_j) when the value of Θ_j (Θ_i) is known given C, and represents the weight that links Θ_i and Θ_j. Once we have learned the undirected structure, a root node must be selected in order to determine the root of the tree by following the structure learned by Algorithm 5.2. Depending on the selected root node and given the undirected tree structure with n nodes, there are n possible resulting directed trees.

- Step 2: We add a class node C to the network structure. We connect this class node to every other node with an arc from C (Fig. 5.4).

- Step 3: Finally, we complete the classification model with the estimation of the parameters for each node given its parent node(s).

Therefore the conditional probability of C given the predictors is

$$p(C = c|\boldsymbol{\Theta} = \boldsymbol{\theta}) \propto p(C = c) f_{\Theta_{root}|C=c}(\theta_{root}|c) \prod_{i=1, i\neq root}^{n} f_{\Theta_i|C=c, Pa_{\Theta_i}}(\theta_i|c, pa_{\theta_i}),$$

where Pa_{Θ_i} is the wC parent of variable Θ_i and Θ_{root} is the root node of the tree.

Similar to the approach used in the rest of the models presented in this Chapter, the maximum a posteriori decision rule is used to determine the predicted class c^*

$$c^* = \arg \max_{c \in \Lambda(C)} p(C = c|\boldsymbol{\Theta} = \boldsymbol{\theta}).$$

5.4 Experimental results

In this Section, we report experiments carried out to show the behaviour of each proposed classification model in Section 5.3. We include the comparison among the four circular classifiers

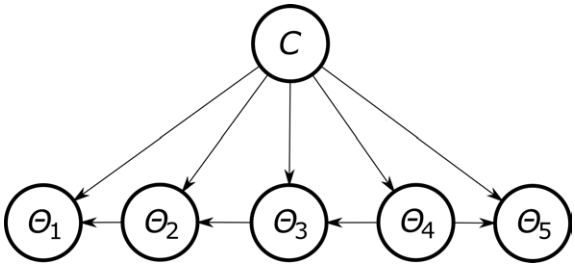

Figure 5.4: wCTAN structure with five nodes, from which $p(c|\boldsymbol{\theta}) \propto p(c) f_{\Theta_1|c,\theta_2}(\theta_1|c,\theta_2) f_{\Theta_2|c,\theta_3}(\theta_2|c,\theta_3) f_{\Theta_3|c,\theta_4}(\theta_3|c,\theta_4) f_{\Theta_4|c}(\theta_4|c) f_{\Theta_5|c,\theta_4}(\theta_5|c,\theta_4)$. The associated tree-structured Bayesian network has Θ_4 as its root node.

and also with the Gaussian TAN classifier (GTAN) for continuous data, with the structure learned with the algorithm in [Geiger and Heckerman, 1994] where predictor variables given the class value are assumed to follow Gaussian distributions.

Simulating data that follows wC distributions is easy and computationally very fast. Given the parameters, the "Circular" R package [Agostinelli and Lund, 2013] simulates wC data by wrapping the simulation of a Cauchy distribution whose location parameter is the same as the wC location parameter and the scale parameter is the negative logarithm of the wC concentration parameter. If the wC concentration parameter is equal to 1, therefore the value of the simulation will be the location parameter, whereas if the concentration parameter is equal to 0, the simulation is performed from a Uniform distribution in $[0, 2\pi)$.

In order to test the algorithms, we enforced dependence between nodes giving values of $|\rho|$ in $[0.5, 1)$. The remaining parameters were assigned randomly to each node with $-\pi < \mu < \pi$ and $0 < \varepsilon < 1$. For each classifier, we simulated 10 datasets each with 1000, 200 and 50 instances and 3, 5, 10, 20, 30, 45, 65, and 100 wC predictor variables and a discrete class variable with 3, 6, 10, 15 and 20 different labels, so we simulated 1200 different datasets for each type of classifier. A 10-fold cross-validation was used to estimate the classification accuracy. Results are shown in Table 5.1.

We also applied the non-parametric Friedman test to detect statistically significant differences among our classification models as a whole set [Friedman, 1937]. When the null hypothesis was rejected, we proceeded with post-hoc tests. We chose the Nemenyi test [Nemenyi, 1963], as suggested by [Demšar, 2006]. The significance level α for all tests was 0.05.

Since multiple classifiers are compared, it is useful to represent the results of the post-hoc tests visually. The graph proposed by Demšar [Demšar, 2006] is a simple diagram to easily represent these results. The top line is the axis on which we plot the average Friedman test ranks of the classifiers. The lowest (best) ranks are to the right, and we therefore consider the classifiers to the right as better. For the comparison results of all classifiers against each other, those that are not significantly different (p-value ≥ 0.05 in the Nemenyi post-hoc test) are connected.

		No. of instances															
		50					200					1000					
		No. of labels					No. of labels					No. of labels					
		3	6	10	15	20	3	6	10	15	20	3	6	10	15	20	
wCNB	3	0.868±0.090	0.544±0.130	0.382±0.171	0.306±0.141	0.252±0.146	0.882±0.046	0.766±0.071	0.597±0.061	0.530±0.058	0.373±0.076	0.882±0.031	0.775±0.056	0.679±0.071	0.605±0.065	0.548±0.046	
	5	0.936±0.067	0.744±0.142	0.486±0.160	0.330±0.155	0.310±0.115	0.957±0.032	0.857±0.050	0.806±0.055	0.672±0.084	0.574±0.074	0.959±0.028	0.886±0.026	0.831±0.032	0.778±0.044	0.737±0.041	
	10	0.996±0.008	0.916±0.087	0.604±0.165	0.490±0.149	0.318±0.187	0.997±0.004	0.976±0.020	0.933±0.039	0.909±0.045	0.832±0.054	0.995±0.009	0.983±0.012	0.974±0.016	0.959±0.016	0.948±0.035	
	20	0.996±0.008	0.976±0.039	0.750±0.177	0.622±0.177	0.413±0.194	0.999±0.001	0.999±0.002	0.996±0.006	0.983±0.017	0.963±0.030	0.999±0.001	0.999±0.001	0.999±0.001	0.997±0.005	0.994±0.009	
	30	0.999±0.001	0.996±0.008	0.832±0.163	0.460±0.209	0.300±0.128	0.999±0.001	0.999±0.001	0.999±0.001	0.995±0.011	0.992±0.016	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.998±0.002	
	45	0.999±0.001	0.999±0.001	0.830±0.154	0.430±0.217	0.250±0.111	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	65	0.999±0.001	0.999±0.001	0.920±0.109	0.500±0.224	0.240±0.132	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	100	0.999±0.001	0.999±0.001	0.860±0.147	0.490±0.271	0.240±0.072	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
wCsNB	3	0.876±0.106	0.628±0.142	0.574±0.141	0.256±0.072	0.238±0.097	0.891±0.044	0.771±0.060	0.607±0.068	0.523±0.070	0.383±0.067	0.895±0.019	0.827±0.022	0.720±0.025	0.627±0.037	0.579±0.035	
	5	0.932±0.062	0.740±0.157	0.432±0.173	0.268±0.105	0.232±0.084	0.949±0.031	0.841±0.054	0.784±0.075	0.644±0.101	0.536±0.098	0.967±0.008	0.914±0.011	0.852±0.023	0.791±0.021	0.738±0.030	
	10	0.948±0.076	0.776±0.142	0.478±0.160	0.276±0.093	0.230±0.117	0.979±0.020	0.910±0.066	0.822±0.102	0.760±0.147	0.646±0.148	0.976±0.005	0.967±0.005	0.944±0.008	0.918±0.007	0.913±0.011	
	20	0.936±0.064	0.824±0.126	0.500±0.170	0.288±0.119	0.252±0.090	0.990±0.014	0.944±0.047	0.872±0.094	0.798±0.120	0.718±0.134	0.994±0.006	0.978±0.018	0.968±0.020	0.948±0.018	0.938±0.025	
	30	0.972±0.033	0.830±0.101	0.560±0.215	0.280±0.110	0.238±0.121	0.990±0.016	0.960±0.046	0.855±0.100	0.887±0.039	0.665±0.174	0.996±0.005	0.981±0.016	0.971±0.019	0.950±0.015	0.953±0.025	
	45	0.988±0.023	0.832±0.121	0.462±0.155	0.292±0.120	0.238±0.106	0.997±0.005	0.980±0.026	0.895±0.120	0.865±0.078	0.792±0.068	0.997±0.002	0.987±0.012	0.976±0.018	0.960±0.012	0.954±0.014	
	65	0.994±0.009	0.856±0.120	0.474±0.150	0.272±0.106	0.262±0.121	0.995±0.013	0.987±0.018	0.960±0.033	0.845±0.172	0.702±0.139	0.998±0.003	0.992±0.009	0.981±0.019	0.965±0.019	0.957±0.02	
	100	0.996±0.008	0.876±0.113	0.584±0.165	0.310±0.103	0.238±0.110	0.995±0.111	0.980±0.027	0.940±0.048	0.832±0.143	0.785±0.136	0.999±0.001	0.996±0.004	0.987±0.012	0.974±0.015	0.963±0.038	
wCsmNB	3	0.880±0.080	0.592±0.136	0.436±0.188	0.266±0.112	0.250±0.088	0.887±0.042	0.778±0.053	0.626±0.082	0.546±0.085	0.403±0.066	0.903±0.010	0.814±0.028	0.715±0.022	0.650±0.039	0.537±0.029	
	5	0.952±0.062	0.820±0.133	0.560±0.158	0.290±0.106	0.271±0.122	0.963±0.027	0.871±0.054	0.929±0.041	0.716±0.076	0.623±0.061	0.965±0.028	0.909±0.019	0.844±0.018	0.828±0.039	0.754±0.051	
	10	0.996±0.008	0.922±0.082	0.768±0.175	0.352±0.156	0.286±0.110	0.997±0.006	0.982±0.016	0.954±0.032	0.940±0.030	0.873±0.056	0.998±0.002	0.987±0.012	0.982±0.020	0.970±0.019	0.960±0.024	
	20	0.999±0.001	0.992±0.014	0.854±0.131	0.486±0.155	0.354±0.113	0.999±0.001	0.998±0.002	0.997±0.003	0.987±0.019	0.979±0.026	0.999±0.001	0.999±0.001	0.999±0.001	0.998±0.002	0.995±0.008	
	30	0.999±0.001	0.996±0.008	0.864±0.129	0.556±0.207	0.360±0.102	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	45	0.999±0.001	0.998±0.004	0.914±0.134	0.632±0.168	0.366±0.127	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	65	0.999±0.001	0.999±0.001	0.980±0.167	0.650±0.137	0.360±0.142	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	100	0.999±0.001	0.999±0.001	0.999±0.001	0.640±0.018	0.420±0.083	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
wCTAN	3	0.868±0.093	0.544±0.130	0.382±0.171	0.230±0.096	0.200±0.100	0.879±0.049	0.754±0.054	0.590±0.060	0.513±0.083	0.378±0.087	0.879±0.025	0.793±0.031	0.716±0.030	0.598±0.040	0.554±0.039	
	5	0.938±0.077	0.774±0.151	0.484±0.142	0.240±0.097	0.160±0.069	0.955±0.034	0.851±0.063	0.774±0.072	0.662±0.078	0.539±0.087	0.980±0.010	0.910±0.014	0.835±0.021	0.776±0.018	0.744±0.024	
	10	0.992±0.017	0.890±0.099	0.694±0.164	0.282±0.170	0.206±0.119	0.995±0.007	0.967±0.023	0.909±0.043	0.872±0.051	0.809±0.061	0.995±0.006	0.981±0.014	0.969±0.015	0.950±0.021	0.935±0.022	
	20	0.999±0.001	0.978±0.031	0.784±0.166	0.396±0.169	0.292±0.135	0.999±0.001	0.997±0.002	0.994±0.010	0.974±0.024	0.951±0.034	0.999±0.001	0.999±0.001	0.999±0.001	0.997±0.004	0.995±0.007	
	30	0.999±0.001	0.999±0.001	0.896±0.123	0.390±0.130	0.282±0.127	0.999±0.001	0.999±0.001	0.999± 0.001	0.997±0.005	0.995±0.011	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	45	0.999±0.001	0.999±0.001	0.848±0.162	0.450±0.199	0.252±0.138	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	65	0.999±0.001	0.999±0.001	0.890±0.134	0.534±0.178	0.246±0.113	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
	100	0.999±0.001	0.999±0.001	0.886±0.149	0.620±0.156	0.244±0.140	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	0.999±0.001	
GTAN	3	0.484±0.162	0.302±0.144	0.214±0.111	0.134±0.082	0.058±0.059	0.541±0.106	0.321±0.079	0.229±0.070	0.180±0.062	0.146±0.052	0.698±0.042	0.387±0.051	0.228±0.041	0.178±0.066	0.119±0.105	
	5	0.588±0.142	0.372±0.131	0.252±0.115	0.154±0.098	0.034±0.028	0.641±0.099	0.643±0.087	0.425±0.079	0.243±0.080	0.225±0.057	0.714±0.021	0.576±0.038	0.467±0.035	0.285±0.041	0.220±0.086	
	10	0.656±0.162	0.460±0.161	0.400±0.148	0.212±0.101	0.008±0.014	0.622±0.091	0.583±0.090	0.521±0.072	0.382±0.070	0.215±0.060	0.803±0.056	0.518±0.048	0.421±0.030	0.325±0.087	0.258±0.090	
	20	0.720±0.152	0.676±0.139	0.488±0.163	0.306±0.162	0.036±0.060	0.777±0.082	0.637±0.054	0.598±0.093	0.437±0.066	0.269±0.089	0.777±0.071	0.637±0.055	0.598±0.059	0.437±0.091	0.372±0.082	
	30	0.774±0.131	0.710±0.148	0.546±0.167	0.340±0.143	0.102±0.106	0.977±0.018	0.805±0.074	0.657±0.059	0.730±0.072	0.677±0.047	0.867±0.026	0.723±0.043	0.679±0.053	0.576±0.059	0.440±0.096	
	45	0.802±0.148	0.778±0.128	0.612±0.158	0.414±0.127	0.144±0.120	0.902±0.057	0.915±0.044	0.855±0.058	0.750±0.058	0.640±0.082	0.940±0.025	0.781±0.021	0.819±0.052	0.688±0.066	0.609±0.096	
	65	0.920±0.076	0.828±0.125	0.706±0.152	0.460±0.174	0.182±0.106	0.951±0.006	0.820±0.020	0.819±0.045	0.867±0.049	0.827±0.047	0.927±0.071	0.874±0.053	0.819±0.046	0.758±0.084	0.681±0.098	
	100	0.816±0.144	0.890±0.083	0.750±0.140	0.504±0.139	0.208±0.141	0.965±0.015	0.935±0.016	0.972±0.018	0.847±0.028	0.710±0.046	0.964±0.019	0.942±0.024	0.834±0.051	0.815±0.074	0.772±0.089	

Table 5.1: Mean ± standard deviation accuracy of wCNB, wCsNB, wCsmNB, wCTAN and GTAN for different number of variables, different number of labels in the class variable for simulated datasets with 50, 200 and 1000 instances.

		Classifiers				
		wCNB	wCsNB	wCsmNB	wCTAN	wCTAN
Number of variables	3	0.735±0.108	**0.755±0.097**	0.754±0.109	0.743±0.108	0.352±0.171
	5	0.866±0.069	0.877±0.069	**0.879±0.066**	0.876±0.074	0.409±0.174
	10	0.976±0.015	0.948±0.021	**0.983±0.011**	0.974±0.018	0.491±0.159
	20	**0.998±0.001**	0.970±0.017	**0.998±0.001**	**0.998±0.001**	0.610±0.122
	30	0.998±0.001	0.976±0.015	**0.999±0.001**	**0.999±0.001**	0.674±0.116
	45	**0.999±0.001**	0.980±0.013	**0.999±0.001**	**0.999±0.001**	0.790±0.091
	65	**0.999±0.001**	0.984±0.013	**0.999±0.001**	**0.999±0.001**	0.824±0.076
	100	**0.999±0.001**	0.989±0.012	**0.999±0.001**	**0.999±0.001**	0.873±0.062

Table 5.2: Mean ± standard deviation accuracy of wCNB, wCsNB, wCsmNB, wCTAN and GTAN classifiers for different number of variables. Results are averaged from the classification performance from Table 5.1 with 3, 6, 10, 15 and 20 different labels with 1000 instances.

5.4.1 Comparison of classification models

In this section, we compare the performance of the wCNB, wCsNB, wCsmNB and wCTAN models, as well as the GTAN algorithm, which ignores the circular nature of the data.

Table 5.2 shows the mean± standard deviation accuracy for each classifier for different number of variables. Each mean± standard deviation accuracy values was obtained from the results of 50 independent 10-fold cross-validation procedures varying the number of labels of the class variable (3, 6, 10, 15 and 20 different labels) with 1000 instances.

The statistical analysis after Friedman test rejection (p-value=0.00035) reveals (Fig. 5.5A) that, varying the number of variables, the best classifiers are wCsmNB, wCTAN and wCNB, with no statistically significant differences among them, whereas the GTAN classifier is the worst, presenting significant differences with respect to the rest of the classifiers and demonstrating that treating circular data as linear-continuous is not effective. The wCsNB also presents statistical differences when compared with the wCsmNB classifier, which outperforms the wCsNB results. Nevertheless, there were no significant differences between the wCsmNB and the rest of the circular classifiers (i.e., wCTAN and wCNB).

Performing the same statistical analysis with 50 and 200 instances yields similar results. The Friedman test is rejected for both (p-value=0.00021 and p-value=0.00004, respectively). The post-hoc analysis displays quite similar results to the 1000 instances one; in both cases, there are no statistically significant differences among the wCsmNB, wCTAN and wCNB classifiers, which are the best. Nevertheless, for 50 and 200 instances, there are no significant differences among GTAN and wCsNB classifiers. Furthermore, as observed for the statistical results with 1000 instances, there are significant differences between the wCsNB and the wCsmNB classifier for the analysis with 50 instances, whereas for 200 instances, statistical differences were seen between the wCsNB classifier and both the wCsmNB and the wCTAN.

We also calculated the mean accuracy for each classifier for different number of labels in the class variable (see Table 5.3). Each mean accuracy value was obtained from the results of 60 independent 10-fold cross-validation procedures varying the number of variables to be

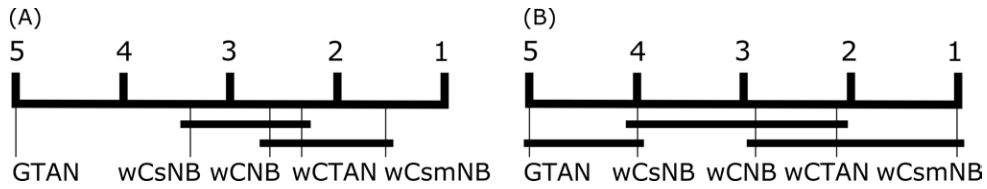

Figure 5.5: Demšar diagrams presenting the statistical comparison among wCNB, wCsNB, wCsmNB, wCTAN and GTAN classification models for synthetic datasets with 1000 instances. Those classifiers that are not connected show differences that are statistically significant (p-value < 0.05). The lowest rank classifiers are to the right side of the graph (i.e., they can be considered the best). (A) Comparison varying the number of labels. (B) Comparison varying the number of variables.

		Classifiers				
		wCNB	wCsNB	wCsmNB	wCTAN	wCTAN
Number of labels	3	0.973±0.042	0.970±0.034	**0.979±0.035**	0.976±0.043	0.709±0.061
	6	0.940±0.083	0.939±0.054	**0.952±0.069**	0.946±0.074	0.526±0.097
	10	0.910±0.118	0.901±0.089	**0.921±0.106**	0.918±0.106	0.384±0.124
	15	0.884±0.146	0.862±0.117	**0.906±0.128**	0.884±0.148	0.311±0.096
	20	0.858±0.167	0.839±0.134	**0.872±0.171**	0.860±0.164	0.253±0.098

Table 5.3: Mean ± standard deviation accuracy of wCNB, wCsNB, wCsmNB, wCTAN and GTAN classifiers for different number of labels. Results are averaged from the classification performance with 3, 5, 10, 20, 30 and 45 different variables with 1000 instances.

used: 3, 5, 10, 20, 30 and 45 with 1000 instances. We do not include the results of the experiments with more than 45 variables due to the high mean accuracy values obtained in most of the classifiers from Table 5.1, which would bias the results.

Since the Friedman test null hypothesis was rejected (p-value=0.00058), we performed the corresponding Nemenyi post-hoc analysis. Statistical test results (Fig. 5.5B) reveal that based on changing the number of labels, the best classifiers are wCsmNB, wCTAN and wCNB, with no statistically significant differences between them. GTAN and wCsNB classifiers are the worst, with no significant differences among them. GTAN shows significant differences with the rest of classifiers, whereas wCsNB only presents significant differences with the wCsmNB classifier.

The analysis for 50 and 200 instances again yielded quite similar results to those obtained for 1000 instances. After Friedman test rejections (p-value=0.0325 for 50 instances, and p-value=0.00066 for 200 instances), post-hoc tests for 200 instances reveal the same statistically significant differences as for 1000 instances, where there is no statistical differences among the wCsmNB, wCTAN and wCNB classifiers, which are the best. For 50 instances, wCsmNB, wCTAN and wCNB are also the best classifiers together with the wCsNB, with no statistically significant differences among them. Likewise, for 1000 instances, GTAN is the worst for the analysis with 50 instances as well as the 200 instances, with no significant differences with the wCsNB classifier.

Therefore, this suggested that it does not seem adequate to use Gaussian distributions

for this kind of data.

5.5 Conclusions and future work

In this chapter, we showed the first set of supervised Bayesian classification models capable of dealing with circular wC predictive variables. We presented four models and their algorithms, designed to perform classification. We demonstrated using synthetic data that these models could perform classification accurately given circular datasets. We also provided evidence of the improvement of the circular classifiers over linear classifiers for datasets of circular nature that follow wC distributions.

We performed statistical comparisons among the classifiers using synthetic data with 50, 200 and 1000 instances. Based on the results, we realised that the wCsmNB, the wCTAN and the wCNB are the best classification models for circular data that follows wC distributions, with no statistically significant differences among them. The GTAN classifier never outperformed any of the wC classifiers and it is therefore never recommended for this type of data.

The models shown in this chapter are limited to no more than bivariate relationships. In future work, we intend to develop multivariate models in order to extend the Bayesian network classifiers for circular data to other more-sophisticated Bayesian network classifiers (like k-dependence Bayesian network classifiers) capable of representing and taking into account multivariate relationships between circular variables.

Chapter 6

Circular-linear dependence measures under Wehrly–Johnson distributions and their Bayesian network application

6.1 Introduction

Several models exist for data consisting of circular and linear observations, most of which focus on a circular-linear regression [Gould, 1969; Fisher and Lee, 1992; SenGupta, 2004]. In addition, Mardia and Sutton [1978] proposed a bivariate distribution that combines the vM distribution with Gaussian distributions on the cylinder. Abe and Ley [2016] proposed the WeiSSVM, a cylindrical distribution based on combinations of the sine-skewed vM distribution and the Weibull distribution. Furthermore, Johnson and Wehrly [1978] presented circular-linear distributions, and proposed a method to obtain a bivariate circular-linear distribution with specified given marginal distributions. In this chapter, we prove that the conditional distributions of a subfamily of Johnson–Wehrly family are well known and mathematically tractable.

Many studies have determined measures for the mutual dependence between linear variables [see Rényi, 1959a,b; Lloyd, 1962, among others], with the MI measure [Shannon, 1948; Cover and Thomas, 2012] being one of the best known. This measure is based on the similarity between the joint density function and the product of its marginal density functions. As mentioned in Chapter 5, we developed the CMI measure [Leguey et al., 2016a]. However, this measure is only applicable for circular variables with marginal distributions that follow wC distributions, and its calculation has to be approximated using numerical methods. To the best of our knowledge, there are no measures of MI for pairs of circular and linear variables. Therefore, in this chapter we propose a circular-linear mutual information measure (CLMI) of the dependence between a circular variable and a linear variable. Furthermore, for the

case when the two variables are in the circular domain, we propose a CMI measure with no constraints on the underlying circular distributions, and that can be expressed in a closed form for a general family of bivariate distributions.

As stated in Chapter 3, one of the main areas in which mutual information measures are applied is that of Bayesian networks. In this chapter, we develop a circular-linear Bayesian network model with a tree structure that captures the relationship between circular and linear variables. The model is based on the proposed CLMI and CMI measures, together with the traditional MI of linear variables and the bivariate distribution proposed by Johnson and Wehrly [1978].

Recently, the study of wind characteristics has become an important field owing to the importance of the location and orientation of wind turbines for profitable wind energy utilization. In this chapter, we use our proposed circular-linear tree-structured Bayesian network to model the relationship between wind speed, wind direction, and other meteorological features from various stations around Europe.

Chapter outline

Section 6.2 reviews the angular-linear Johnson–Wehrly bivariate distribution, and shows that its conditional distributions are tractable and well known. Section 6.3 discusses the proposed CMI and CLMI measures. Section 6.4 applies the measures presented in Section 6.3, and presents the proposed circular-linear tree-structured Bayesian network model, as well as its evaluation in synthetic domains. Section 6.5 compares the proposed model to a Gaussian Bayesian network model and a discrete Bayesian network model over a real-world meteorological data set recorded from meteorological stations located in Europe. Lastly, Section 6.6 concludes the paper and discusses possible avenues for future work.

6.2 Circular-linear distribution of Johnson and Wehrly

In this section, we review the method proposed by Johnson and Wehrly [1978] to obtain angular-linear bivariate distributions with arbitrary marginal distributions.

After reviewing the general form of the Johnson–Wehrly angular-linear distribution, we propose a subfamily in which the conditionals and marginals are mathematically tractable.

6.2.1 Definition

Let $f_\Theta(\theta)$ and $f_X(x)$ be probability density functions on the circle and on the line, respectively. Suppose that $F_\Theta(\theta)$ is the cumulative distribution function (CDF) of $f_\Theta(\theta)$ defined with respect to a fixed, arbitrary origin, and that $F_X(x)$ is the CDF of $f_X(x)$. Then, the distribution of Johnson and Wehrly [1978] is defined by the density

$$f(\theta, x) = 2\pi g\left(2\pi F_\Theta(\theta) - 2\pi q F_X(x)\right) f_\Theta(\theta) f_X(x), \tag{6.1}$$

where $0 \leq \theta < 2\pi$, $x \in \mathbb{R}$, $q \in \{-1, 1\}$ decides the positive or negative association between the two variables, and $g(.)$ is a density on the circle.

Let a random vector (Θ, X) have the density given by Equation (6.1). Then the marginal distribution of Θ has the density $f_\Theta(\theta)$ and CDF $F_\Theta(\theta)$, while the marginal distribution of X has the density $f_X(x)$ and CDF $F_X(x)$. However the conditional distributions are not tractable in general.

6.2.2 Conditionals

Let a random vector (Θ, X) follow the distribution given by Equation (6.1). Then, changing the variables $U = 2\pi F_\Theta(\Theta)$ and $V = 2\pi F_X(X)$, the density function of (U, V) can be expressed as

$$f(u, v) = \frac{1}{2\pi} g(u - qv),$$

where $g(.)$ is a density on the circle and $q \in \{-1, 1\}$.

We propose the following subfamily of the family given by Equation (6.1) which has tractable conditional distributions.

Theorem 6.1. Let a random vector (Θ, X) follow the distribution given by Equation (6.1) with $g(.)$ being the wC density given by Equation (2.4) with location parameter μ_g and concentration parameter ε_g. Assume that $U = 2\pi F_\Theta(\Theta)$ and $V = 2\pi F_X(X)$. Then

$$U|X = x \sim wC(2\pi q F_X(x) + \mu_g, \varepsilon_g) \quad \text{and} \quad V|\Theta = \theta \sim wC(q(2\pi F_\Theta(\theta) - \mu_g), \varepsilon_g).$$

In particular, if $\Theta \sim wC(\mu_\theta, \varepsilon_\theta)$, $X \sim N(\iota_x, \sigma_x^2)$ and $F_\Theta(\theta) = \int_0^t f_\Theta(t)dt$, then, it holds that

$$\Theta|X = x \sim wC(\mu, \varepsilon),$$

where $\mu = arg(\hat{\phi}_{\theta|x})$, $\varepsilon = |\hat{\phi}_{\theta|x}|$,

$$\hat{\phi}_{\theta|x} = \frac{\varepsilon_g e^{(i(2\pi q F_X(x) + \mu_g - \nu))} + \varepsilon_\theta e^{(i\mu_\theta)}}{1 + \varepsilon_g \varepsilon_\theta e^{(i(2\pi q F_X(x) + \mu_g - \mu_\theta - \nu))}},$$

and $\nu = \arg\{(1 - \varepsilon_\theta e^{(-i\mu_\theta)})/(1 - \varepsilon_\theta e^{(i\mu_\theta)})\}$.

See A.1 for the proof.

Therefore, the conditional of the circular variable, given the linear variable, follows a known and tractable distribution (i.e. a wC distribution (Equation (2.4))).

The conditional $X|\Theta = \theta$ itself does not follow any well-known distribution. However, Theorem 6.2 implies that

$$2\pi\, \Phi\left(\frac{X - \iota_x}{\sigma_x}\right) |\Theta = \theta \sim wC(q(2\pi F_\Theta(\theta) - \mu_g), \varepsilon_g),$$

where Φ denotes the CDF of the standard Gaussian distribution $N(0, 1)$, namely, $\Phi(x) = \int_{-\infty}^{x} \phi(t)dt$, where ϕ is the the Gaussian density with $\iota_x = 0$ and $\sigma_x = 1$. Since it is easy to

evaluate the CDF of the standard Gaussian distribution numerically, numerical calculations associated with the conditional distribution of X given $\Theta = \theta$ can be conducted efficiently.

6.3 Measures of mutual dependence

Mutual dependence measures between two linear variables have been studied at length, including the works of [Rényi, 1959b,a; Lloyd, 1962], among many others. In the case of linear data, one of the best-known measures is mutual information [Shannon, 1948; Cover and Thomas, 2012], which determines the similarity between the joint density and the product of its marginal densities. For circular data, the CMI was developed recently by [Leguey et al., 2016a]. However, the CMI is defined for circular variables only, each of which follows a wC distribution and has a joint density that follows a bivariate wC distribution.

In this section, we redefine the CMI such that the measure can be used for any circular variables. Then, we present a closed-form expression for the CMI for the general family of bivariate circular distributions. This study is also the first to propose a mutual-information measure for circular and linear variables, which we call CLMI.

6.3.1 Circular mutual information

Because the CMI presented in [Leguey et al., 2016a] has to be approximated using numerical methods, we present a closed-form expression for CMI that does not rely on the underlying distribution of the circular variables.

Let Θ, Ψ be a pair of circular variables. Then, the CMI between Θ and Ψ is defined by

$$\mathrm{CMI}(\Theta, \Psi) = \int_0^{2\pi} \int_0^{2\pi} f(\theta, \psi) \log \left\{ \frac{f(\theta, \psi)}{f(\theta) f(\psi)} \right\} d\psi d\theta, \tag{6.2}$$

where $f_\Theta(\theta)$ is the marginal density of Θ, $f_\Psi(\psi)$ is the marginal density of Ψ, and $f(\theta, \psi)$ is the joint density of (Θ, Ψ).

Following a similar method to that in Johnson and Wehrly [Johnson and Wehrly, 1978], Wehrly and Johnson presented a general family for circular variables in [Wehrly and Johnson, 1980]. Here we consider the following subfamily with the joint density function

$$f(\theta, \psi) = 2\pi\delta(2\pi F_\Theta(\theta) - 2\pi q F_\Psi(\psi)) f_\Theta(\theta) f_\Psi(\psi), \tag{6.3}$$

where $0 \leq \theta, \psi < 2\pi$, $f_\Theta(\theta)$ and $f_\Psi(\psi)$ are any probability density functions on the circle, $F_\Theta(\theta)$ and $F_\Psi(\psi)$ are the CDFs of $f_\Theta(\theta)$ and $f_\Psi(\psi)$, respectively, $q \in \{1, -1\}$ decides the positive or negative association between the two variables, and $\delta(\nu)$ is the wC probability density proposed in [Kato and Jones, 2015], with μ_δ as the location parameter and ε_δ as the concentration parameter.

Assume that a random vector (Θ, Ψ) has the distribution given by Equation (6.3). Then it holds that the marginal distribution of Θ has the density $f_\Theta(\theta)$ and the CDF $F_\Theta(\theta)$ and that the marginal distribution of Ψ has the density $f_\Psi(\psi)$ and the CDF $F_\Psi(\psi)$. The bivariate

wC distribution from Equation (5.1) proposed in [Kato and Pewsey, 2015] is a special case of the Wehrly and Johnson general family [Wehrly and Johnson, 1980].

Theorem 6.2. Let (Θ, Ψ) have the distribution given by Equation (6.3). Then, the CMI between Θ and Ψ defined in Equation (6.2) is given by

$$\text{CMI}(\Theta, \Psi) = 2\pi \int_0^{2\pi} \delta(t) \log\{\delta(t)\} dt.$$

In particular, if δ is the wC density given in Equation (2.4), with location parameter μ_δ and concentration parameter ε_δ, then

$$\text{CMI}(\Theta, \Psi) = -\log(1 - \varepsilon_\delta^2). \tag{6.4}$$

The proof of this theorem is given in A.2.

This CMI measure is expressed in a closed form. Therefore, it is computationally very fast and can be used for pairs of variables that fit any circular distribution that allows the calculation of the CDF.

In order to estimate ε_δ in Equation (6.4) which is unknown, we use an estimate from [Kato and Pewsey, 2015] based on a sample of size $N : \{(\theta_1, \psi_1), .., (\theta_N, \psi_N)\}$, given by

$$\hat{\varepsilon}_\delta = \frac{1}{N} \left(\left| \sum_{j=1}^{N} e^{\left(i(2\pi \hat{F}_\Theta(\theta_j) - (2\pi \hat{F}_\Psi(\psi_j))\right)} \right| - \left| \sum_{j=1}^{N} e^{\left(i((2\pi \hat{F}(\theta_j) + (2\pi \hat{F}(\psi_j))\right)} \right| \right),$$

where $\hat{F}_\Theta(\theta_j)$ and $\hat{F}_\Psi(\psi_j)$ denote the empirical CDFs of Θ and Ψ in the sample, respectively.

Considering the particular case where both variables follow wC distributions, as described in Equation (2.4), the estimate of ε_δ simplifies to the correlation parameter ρ between Θ and Ψ described in Equation (5.5).

6.3.2 Circular-linear mutual information

Following the development of the CMI, we next present the CLMI, which allows a closed-form expression for the mutual-information measure for a general family of distributions.

Let Θ be a circular variable, and let X be a linear variable. Then, the CLMI between Θ and X is defined by

$$\text{CLMI}(\Theta, X) = \int_{-\infty}^{\infty} \int_0^{2\pi} f(\theta, x) \log \left\{ \frac{f(\theta, x)}{f_\Theta(\theta) f_X(x)} \right\} d\theta dx, \tag{6.5}$$
$$0 \leq \theta < 2\pi, \; x \in \mathbb{R},$$

where $f_\Theta(\theta)$ is the marginal density of Θ, $f_X(x)$ is the marginal density of X, and $f(\theta, x)$ is the joint density of (Θ, X).

Theorem 6.3. Let (Θ, X) have the distribution density of Equation (6.1). Then, the CLMI between Θ and X, defined in Equation (6.5), is given by

$$\text{CLMI}(\Theta, X) = 2\pi \int_0^{2\pi} g(t) \log\{g(t)\} dt.$$

In particular, if g is the wC density, as in Equation (2.4), with location parameter μ_g and concentration parameter ε_g, then

$$\text{CLMI}(\Theta, X) = -\log(1 - \varepsilon_g^2). \tag{6.6}$$

The proof of this theorem is given in A.3.

Because ε_g in Equation (6.6) is unknown in the usual settings, we need to estimate ε_g in order to estimate the value of CLMI. Similarly to the CMI, an estimate of ε_g based on a sample of size $N : \{(\theta_1, x_1), .., (\theta_N, x_N)\}$ is given by

$$\hat{\varepsilon}_g = \frac{1}{N} \left| |\sum_{j=1}^{N} e^{\left(i(2\pi \hat{F}_\Theta(\theta_j) - 2\pi \hat{F}_X(x_j))\right)}| - |\sum_{j=1}^{N} e^{\left(i(2\pi \hat{F}_\Theta(\theta_j) + 2\pi \hat{F}_X(x_j))\right)}| \right|,$$

where $\hat{F}_\Theta(\theta)$ and $\hat{F}_X(x)$ are the estimated distribution functions of Θ and X, respectively. It can be seen that $\hat{\varepsilon}_g$ is a method of moments estimator of ε_g by noting that $|\text{E}[e^{(i(2\pi F_\Theta(\Theta) - 2\pi F_X(X)))}]| = \varepsilon_g$ and $|\text{E}[e^{(i(2\pi F_\Theta(\Theta) + 2\pi F_X(X)))}]| = 0$. Similarly to the CMI, the CLMI for the Johnson–Wehrly family of distributions provided by Equation (6.1) is computationally very fast because it is expressed in closed form. In addition, it can be used to compute the mutual-information measure between circular and linear variables without needing any assumptions on their marginal distributions.

If we consider the particular case where $\Theta \sim wC(\mu_\theta, \varepsilon_\theta)$ and $X \sim N(\iota_x, \sigma_x)$, then the estimate of ε_g simplifies to

$$\hat{\varepsilon}_g = \frac{1}{N} \left| |\sum_{j=1}^{N} e^{(i(\Phi_j - \tau_j))}| - |\sum_{j=1}^{N} e^{(i(\Phi_j + \tau_j))}| \right|,$$

where $\Phi_j = 2\pi \hat{F}_\Theta(\theta_j) = 2 \arctan\left(\frac{1+\hat{\varepsilon}_\theta}{1-\hat{\varepsilon}_\theta} \tan\left(\frac{\theta_j - \hat{\mu}_\theta}{2}\right)\right)$ and $\tau_j = 2\pi \hat{F}_X(x_j) = \pi \,\text{erf}\left(\frac{x_j - \hat{\iota}_x}{\hat{\sigma}_x \sqrt{2}}\right)$, where $\hat{\varepsilon}_\theta$ and $\hat{\mu}_\theta$ are as in Section 2.1, $\hat{\iota}_x$ and $\hat{\sigma}_x$ are as in Section 2.2, and erf is the Gauss error function.

6.4 Circular-linear tree-structured Bayesian network learning

In this section, we apply the CLMI and the CMI measures presented in Section 6.3.

We use score-search (Section 3) for the structure learning in a Bayesian network with a tree-structure. The importance of this application is that it allows the use of both linear and circular variables to develop this kind of Bayesian network model.

Our proposed algorithm (Algorithm 6.1) is based on the algorithm introduced by Chow and Liu [Chow and Liu, 1968] to find a maximum weight spanning tree structure. The weight between a pair of variables is measured as the CLMI, CMI, or MI between them, depending on the nature of the pairwise joint densities. Note that the range of values for the mutual-information measures is $[0, \infty)$.

Algorithm 6.1 Adaptation of the Chow–Liu algorithm

1: Given $\Theta_1, \Theta_2, ..., \Theta_n$ circular random variables and $X_1, X_2, ..., X_m$ linear random variables,
   (i) Estimate the parameters of all marginals $f(\theta_i)$ and of $f(x_i)$.
   (ii) Estimate the dependence parameters ε_δ, γ, and ε_g of the joint density functions $f(\theta_i, \theta_j)$, $f(x_i, x_j)$, and $f(\theta_i, x_j)$, respectively, for all pairs of variables based on the estimated parameters of the marginals.
2: Using these distributions, compute all values of $\text{CLMI}(\Theta_i, X_j)$, $\text{CMI}(\Theta_i, \Theta_j)$, and $\text{MI}(X_i, X_j)$ (i.e. the $(n+m)(n+m-1)/2$ edge weights), and order them.
3: Assign the largest two edges to the undirected tree to be represented.
4: Examine the next-largest edge, and add it to the current tree, unless it forms a loop, in which case discard it and examine the next largest edge.
5: Repeat step 4 until $n+m-1$ edges have been selected (and the spanning undirected tree is complete).
6: Choose a root node and follow the structure to create the maximum weight spanning directed tree structure.

Given the undirected tree structure from Algorithm 6.1, there are $n+m$ possible trees, depending on the selected root node. Note that if the estimate of the parameters method in Step 1 is performed via MLE, the generated maximum weight spanning directed tree is also a maximum likelihood tree. Otherwise, while we can confirm that the resulting tree is a maximum spanning tree, we cannot ensure that it is a maximum likelihood tree.

6.4.1 Experimental results

Owing to the properties explained in Sections 2.2.3.2, 5.2 and those well-known from the Gaussian distribution [see Johnson et al., 1970; Tong, 1990; Kotz et al., 2004, for good reviews] that make the structure-learning phase easier, the data used in the experiments are simulated from Gaussian distributions and wC distributions. We generate 12250 different simulated structures: 250 different structures for every combination of 3, 5, 7, 10, 15, 20, and 30 linear Gaussian variables with 3, 5, 7, 10, 15, 20, and 30 circular wC variables (i.e. 49 combinations of variables). From each of these structures we simulate a data set with $N = 500$ instances, where the parameters of the variables are assigned based on the uniform distributions on the following ranges: $-1000 < \iota < 1000$ and $0 < \sigma < 100$ for the Gaussian marginal distributions and $-\pi < \iota < \pi$ and $0 < \varepsilon < 1$ for the wC marginal distributions. In addition, to enforce the dependence between the parent and children nodes of the network, we assign a correlation parameter between them as a uniform random value $0.5 < |\rho| < 1$, with

Table 6.1: Mean accuracy ± standard deviations of the simulation results for the circular-linear tree-structured Bayesian network for different combinations of circular and linear variables.

		Linear variables						
		3	5	7	10	15	20	30
Circular variables	3	0.710±0.191	0.723±0.185	0.703±0.185	0.691±0.191	0.697±0.194	0.710±0.192	0.689±0.208
	5	0.744±0.173	0.741±0.169	0.774±0.166	0.749±0.172	0.736±0.169	0.749±0.170	0.766±0.163
	7	0.804±0.146	0.780±0.144	0.791±0.139	0.796±0.140	0.791±0.136	0.811±0.137	0.788±0.147
	10	0.830±0.119	0.828±0.123	0.825±0.120	0.808±0.125	0.831±0.115	0.817±0.123	0.837±0.122
	15	0.858±0.096	0.854±0.090	0.866±0.094	0.867±0.092	0.863±0.096	0.860±0.095	0.863±0.095
	20	0.879±0.077	0.884±0.075	0.886±0.079	0.876±0.084	0.879±0.078	0.882±0.079	0.878±0.084
	30	0.900±0.061	0.901±0.060	0.910±0.067	0.903±0.062	0.899±0.060	0.900±0.065	0.904±0.060

$q = 1$. Then, we apply the structure-learning algorithm proposed in Algorithm 6.1 over the generated data sets. To measure the accuracy of the results, we compare the misplaced arcs of the learned structure to the initial simulated structure. The results of these experiments are displayed in Table 6.1, where the results of the 49 combinations of variables are shown as mean accuracy ± standard deviations. We refer to accuracy as the percentage of non-misplaced arcs (i.e. how accurate a replicated structure is when compared against the original structure).

To better understand the algorithm behaviour when varying the number of circular and linear variables, we apply the Friedman test [Friedman, 1937] (further information in Section 5.4) (significance level $\alpha = 0.05$). Here, the null hypothesis of equality between the sets was rejected (p-value ≤ 0.05). Therefore, we proceeded with the Nemenyi post-hoc test [Nemenyi, 1963](further information in Section 5.4) (significance level $\alpha = 0.05$) to compare the sets of results with each other.

Figure 6.1 provides the statistical test results using Demšar's diagram [Demšar, 2006] (further information in Section 5.4).

Analysing Table 6.1 and Figure 6.1(a), which shows the results of changing the number of linear variables, we find that the Friedman test is not rejected (p-value = 0.61). This implies that there is no statistically significant difference between the experimental results when the number of linear variables varies. Nevertheless, Figure 6.1(b), in which the number of circular variables varies, shows that the Friedman test is rejected (p-value = 0.000018). Furthermore, the accuracy increases as the number of circular variables increases. The case of 30 circular variables is the most accurate, and the case with 3 circular variables is the least accurate. In addition, conducting Nemenyi's post-hoc tests for every pair of cases, we find no statistically significant differences between consecutive trios of cases (i.e. 3-5-7, 5-7-10, 7-10-15, 10-15-20, and 15-20-30).

6.5 Real example

In this section, we apply the proposed structure learning for our circular-linear Bayesian network to a meteorological data set. The data include circular and linear measurements collected from seven wind stations located in Europe: Baltic Sea in Poland, Black Sea in Ro-

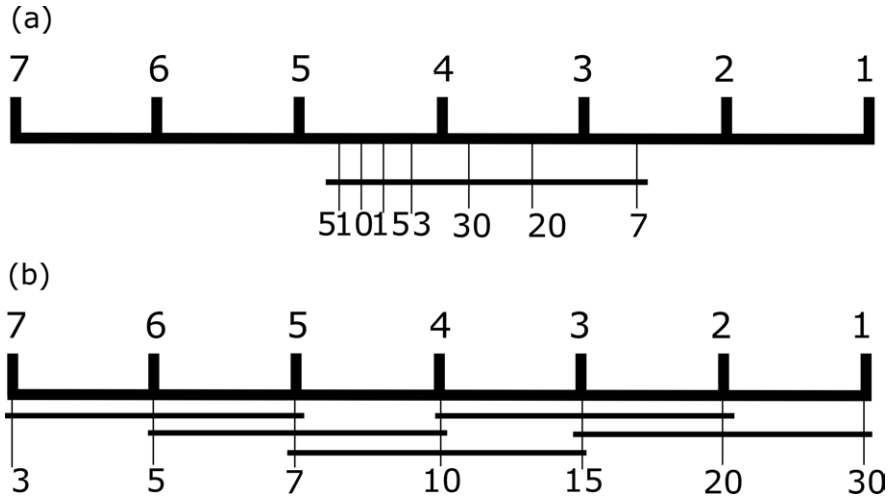

Figure 6.1: Demšar diagram to compare the experimental results by varying the number of (a) linear variables or (b) circular variables.

mania, Dwejra in Malta, Hegyhatsal in Hungary, Lampedusa in Italy, Pallas-Sammaltunturi in Finland, and the M Ocean station in Norway (Figure 6.2). Each station records wind direction and wind speed measurements. In addition, relative humidity, atmospheric temperature, and atmospheric pressure are recorded at the Lampedusa station. These data are available at the World Data Centre for Greenhouse Gases[1] (WDCGG). In our data set, we consider the records for the period 16 July 1992 to 29 December 2005. Since some stations report more than one record per day, we calculate the mean value per day for linear measurements and the mean direction per day for circular measurements. Thus, the data set has $n+m = 7+10 = 17$ variables (i.e. wind speed and wind direction from every station, as well as the additional three measures from the Lampedusa station) and $N = 3301$ instances.

Table 6.2 shows the names of the variables, the numbers of samples of the variables, and the parameter estimates of the marginal distributions. Note that some circular variables are close to circular uniformity because $\hat{\varepsilon} \simeq 0.05$. Owing to the nature of the data, which are recorded over a long period, the circular data show low concentration parameters. The highest concentration parameter is in the Dwejra station in Malta, where $\hat{\varepsilon} = 0.39$, with $\hat{\mu} = -0.77$ (about 311 degrees). Note that Lampedusa station, which is close to Dwejra station, shows a similar mean wind direction of $\hat{\mu} = -0.58$ (about 327 degrees) and a concentration parameter of $\hat{\varepsilon} = 0.20$. With regard to wind speed linear variables, note that the highest values are shown in the M Ocean station and Baltic sea station, both of which are located in the sea or on the coast of the Northern Europe territory.

The circular-linear tree-structured Bayesian network model (Figure 6.3) reveals the conditional relationship between the variables measured from each meteorological station. Note that most Lampedusa station nodes are connected to Dwejra station nodes. This is because these two stations are geographically close together, as previously mentioned. The figure

[1]WDCGG URL: `http://ds.data.jma.go.jp/gmd/wdcgg/`

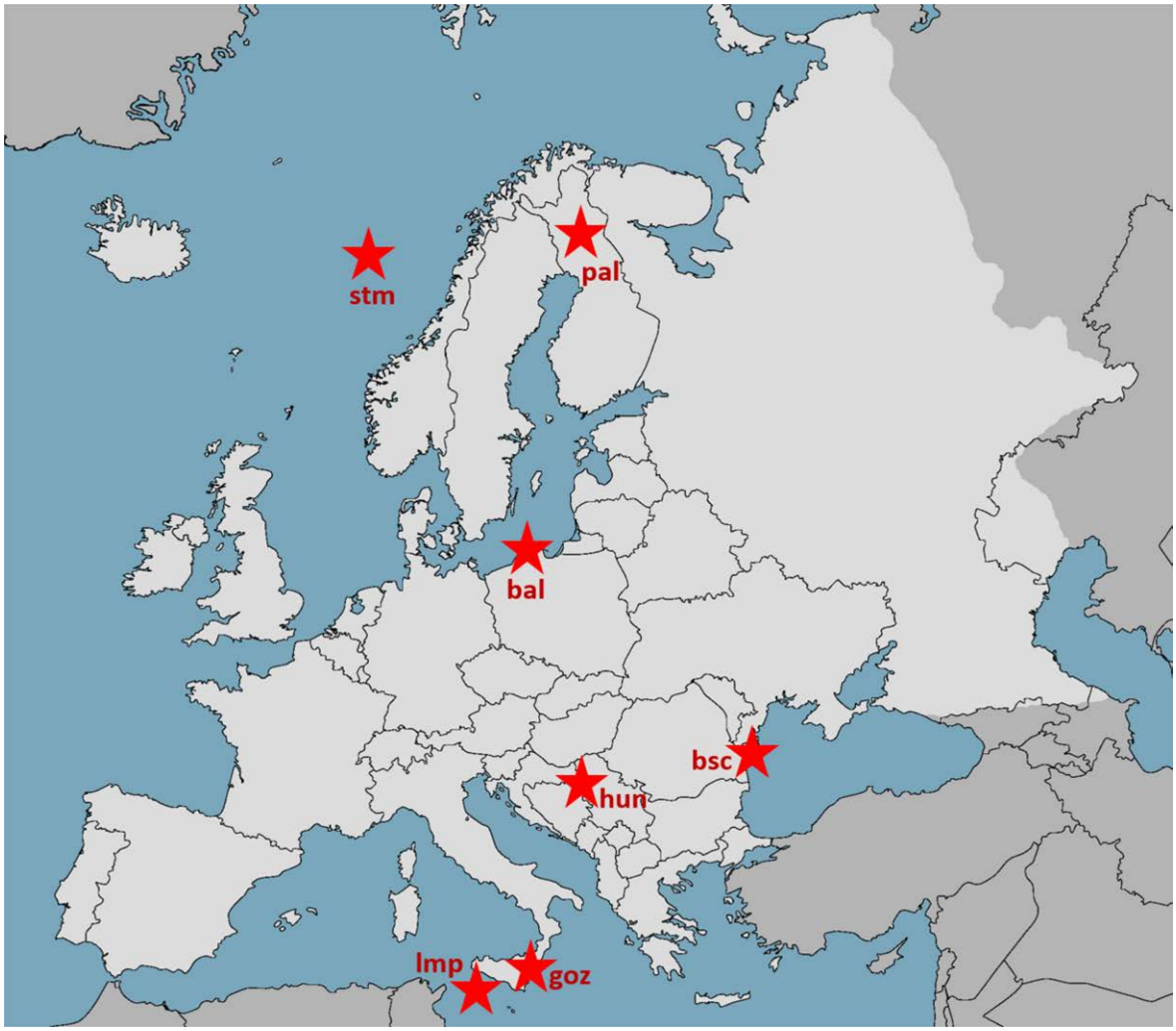

Figure 6.2: European locations of the meteorological stations from the WDCGG data set

Table 6.2: Station name, variable described, variable used in the model, number of non-missing cases (N_i), circular mean $\hat{\mu}$ or linear mean $\hat{\iota}$ (where applicable), unit of measure, concentration $\hat{\varepsilon}$ or standard deviation $\hat{\sigma}$ (where applicable), and type of variable (C: Circular; L: Linear) for the 17 numeric variables of the WDCGG data set. The circular variables range from $-\pi$ to π.

Station	Variable	Name	N_i	$\hat{\mu}/\hat{\iota}$	Units	$\hat{\varepsilon}/\hat{\sigma}$	Type
M Ocean	Wind direction	stmWD	1288	-1.86	*radians*	0.06	C
	Wind speed	stmWS	1288	8.93	m/s	3.91	L
Pallas-Sammaltunturi	Wind direction	palWD	150	-2.76	*radians*	0.18	C
	Wind speed	palWS	150	6.4	m/s	3.21	L
Lampedusa	Wind direction	lmpWD	446	-0.58	*radians*	0.20	C
	Wind speed	lmpWS	446	6.62	m/s	3.94	L
	Relative humidity	lmpRH	446	56.4	%	27.8	L
	Atmospheric pressure	lmpAP	446	1010	*hPa*	6.29	L
	Atmospheric temp.	lmpAT	446	19.59	*Celsius*	5.26	L
Hegyhatsal	Wind direction	hunWD	557	-0.67	*radians*	0.07	C
	Wind speed	hunWS	557	3.82	m/s	3.20	L
Dwejra	Wind direction	gozWD	157	-0.77	*radians*	0.39	C
	Wind speed	gozWS	157	2.79	m/s	2.17	L
Black Sea	Wind direction	bscWD	550	0.47	*radians*	0.22	C
	Wind speed	bscWS	550	4.73	m/s	2.56	L
Baltic Sea	Wind direction	balWD	1169	-1.77	*radians*	0.21	C
	Wind speed	balWS	1169	9.93	m/s	4.90	L

also shows other geographically close node connections, such as the arc between the Pallas-Sammaltunturi and M Ocean stations, both located at the Northern Europe territory, and the arc between the Black Sea and the Hegyhatsal stations, which are the two Eastern Europe stations. Therefore, it seems that our circular-linear model is capturing the dependence relationships between the variables of the data set properly.

The Schwarz Bayesian information criterion (SBIC) [Schwarz, 1978] is a variation of the BIC, presented in Section 3.3, where the lowest value is preferred. It is based on the likelihood function with an overfitting penalty, and is defined as

$$\text{SBIC} = -2ln\hat{L} + ln(N)w,$$

where $\hat{L}$ is the likelihood function value, N is the sample size, and w is the number of parameters to be estimated in the model.

We use the SBIC to compare our model to a Gaussian Bayesian network model, where we assume that all variables follow Gaussian continuous distributions. We also compare our model to a discrete Bayesian network model using the discretization of variables. This discretization is carried out by considering each variable as linear, and then creating 10 partitions of equal width (these vary for each variable, depending on their corresponding domain). Note that for those cases where the variables are circular, the linear domain is considered to be between $-\pi$ and π.

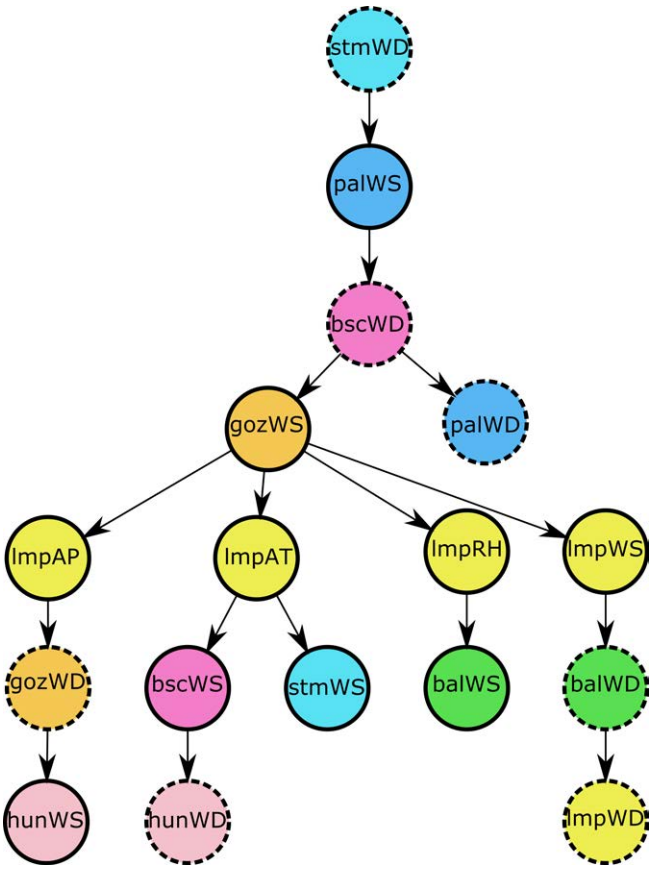

Figure 6.3: Circular-linear tree-structured Bayesian network for the WDCGG meteorological data set. The names of the variables are shown in Table 6.2. The selected root node is the wind direction at the M Ocean station. Dashed border node lines indicate circular variables, while solid border node lines indicate linear variables. Nodes with the same colour are recorded at the same station. Nodes with similar colour tones are located close to each other geographically.

Table 6.3: SBIC comparison between the circular-linear Bayesian network model, Gaussian Bayesian network model, and discrete Bayesian network model for the WDCGG data set.

Bayesian network Model	SBIC
Circular-Linear	$-5.9197 * 10^{192}$
Gaussian	$-2.0896 * 10^{169}$
Discrete	$-3.9851 * 10^{4}$

Comparing the SBIC values in Table 6.3, we observe that our circular-linear model clearly outperforms the other two models, which ignore the circular nature of the circular variables and treat them as (linear) continuous or discrete variables.

6.6 Conclusions

Circular data are often observed together with linear data in the sciences. In this chapter, we showed that the subfamily of Johnson–Wehrly bivariate distributions has tractable properties, such as well-known marginals and conditionals and a closed-form expression for the estimators of the parameters. We presented a CLMI measure, which measures the mutual dependence between a circular variable and a linear variable by determining the similarity between the joint density and the product of their marginal densities. We also extended the definition of the CMI measure. We showed that the CLMI and CMI can be expressed in a simple and closed form for our distributions for circular-linear data and bivariate circular data, respectively.

In addition, we described experimental results that illustrate how to use these measures (i.e., the CLMI and CMI) with the well-known MI between linear variables. To the best of our knowledge, this study is the first to develop a circular-linear tree-structured Bayesian network model that can capture the dependence between any possible pair of linear and circular variables.

Then, we applied our tree-structured Bayesian network to a real data set in order to model the relationships between circular and linear measurements recorded at seven meteorological stations located in Europe. Here, we observed that the proposed model captures well the strong dependence between variables recorded in geographically close stations, and outperforms other models, which assume that all variables are Gaussian or discrete.

Working with a combination of circular and linear statistics is a non-trivial task. Applications within Bayesian networks and machine learning research for graphical models open a challenging field. As future work, we intend to adapt the proposed circular-linear graphical model for supervised classification models. In addition, dropping the dimension constraint (of one parent) in our model would be another interesting path to explore in order to extend this model to a more general Bayesian network case.

Part IV

CONTRIBUTIONS TO NEUROSCIENCE

Chapter 7

Dendritic branching angles of pyramidal cells across layers of the juvenile rat somatosensory cortex

7.1 Introduction

As we explained in Chapter 4, pyramidal neurons represent the most abundant neuronal type in the cerebral cortex. Their dendritic spines constitute the major postsynaptic elements of cortical excitatory synapses and are fundamental to memory, learning, and cognition [Spruston, 2008; Yuste, 2010; DeFelipe, 2015]. Thus, our understanding of the synaptic organization of the neocortex largely depends on the available knowledge regarding pyramidal cells. To date several studies have shown that pyramidal cells sampled from different areas of different species, including rodents and primates, present quantitative differences in the size and complexity of the dendritic arbor and the density of spines [Elston et al., 2001; Jacobs et al., 2001; Elston, 2003; Benavides-Piccione et al., 2006]. Also differences between layers and age have been reported in various species [Larkman, 1991; Petanjek et al., 2007; Oberlaender et al., 2011; Benavides-Piccione et al., 2012]. These variations reflect differences in cortical information processing. For example, different branch structures are responsible for different forms of processing within the dendritic tree before input potentials arrive at the soma [reviewed in Stuart and Spruston, 2015]. Therefore, there may be a greater potential for compartmentalization in areas that contain highly branched pyramidal cells than in areas with less branched cells [reviewed in Elston, 2003].

Previous studies have identified several rules that seem to be common in dendritic geometry. For example, it has been proposed that geometric theory predicts bifurcations in minimal wiring cost trees [Cuntz et al., 2008; Wen et al., 2009; Cuntz et al., 2010; van Pelt and Uylings, 2011; Cuntz et al., 2012; Kim et al., 2012]. Also, it has been described that dendrites usually branch when they are close to the soma to produce short segments, whereas the segments that do not branch spread away from the soma [Samsonovich and Ascoli, 2003;

López-Cruz et al., 2011]. These studies have shown that segment orientation is mainly controlled by the orientation of the previous segments and that dendritic trees tend to first spread rapidly when they are close to the soma and then, once they have reached a minimum size, grow straight away from the soma. Additionally, the first bifurcation of a particular basal tree is the widest, and subsequent bifurcations become progressively narrower [López-Cruz et al., 2011; Bielza et al., 2014]. Moreover, the final bifurcation of a particular cortical region is rather similar, regardless of the branch order of the dendrite [Bielza et al., 2014]. In this chapter, we analyse the geometry of pyramidal cell basal arbors in different cortical layers of the juvenile Wistar rat somatosensory cortex to determine if the above rules are applicable to the different cortical layers. We used Wistar rats at postnatal day 14 since we intended to integrate these data with other anatomical, molecular and physiological data that have already been collected from the same cortical region of the P14 Wistar rats. The final goal is to create a detailed, biologically accurate model of circuitry through layers II - VI in the primary somatosensory cortex, within the framework of the BBP (see Section 4.3 for further details of the BBP).

The research included in this chapter has been published in Leguey et al. [2016b].

Chapter outline

Section 7.2 includes information about the dendritic dataset, an overview of the used methods for this study and information about the supplementary material not presented in this dissertation. In Section 7.3 the results of the study are reported. Finally, Section 7.4 discusses these results as well as some future work.

7.2 Materials and methods

7.2.1 Supplementary material

A website[1] has been set up containing supporting information from the experiments reported in this section. This includes the Supplementary Figures 1-7 with the remaining cases not shown in the figures of this chapter, and the Supplementary Tables S1-S16 with the experiment sample sizes and the performed statistical tests results.

7.2.2 Data

A set of 288 3D pyramidal neurons from six different layers of the 14-day-old (P14) rat hind limb somatosensory (S1HL) neocortex was used for the analysis. Current methodological limitations restrict us to the study of either the complete basal arbors (horizontal sections) or truncated apical and basal arbors (coronal sections). For the sake of consistency with our previous studies, we opted to study the basal dendrites. Thus, pyramidal neurons were intracellularly injected in horizontal sections to allow the study of complete basal dendritic

[1] Supplementary material url: http://cig.fi.upm.es/suppmaterialLegueyetal

arbors. Briefly, cells in layers II, III, IV, Va, Vb and VI were individually injected with Lucifer Yellow, which was applied by continuous current until the distal tips of each dendrite fluoresced brightly. Following injections, the sections were processed with an antibody to Lucifer Yellow to visualize the complete morphology of the cells (Figure 7.1A, 7.1B). Only neurons that had an unambiguous apical dendrite and whose basal dendritic tree was completely filled and contained within the section were included in the analysis (48 cells from each layer; 6 cells per layer, 6 layers, 8 animals). The Neurolucida package (MicroBrightField[2]) was used to three-dimensionally trace the basal dendritic arbor of each pyramidal cell (Figure 7.1C). Reconstruction of the same neurons has been used previously in another study for different purposes [Rojo et al., 2016]. Further information regarding tissue preparation, injection methodology, immunohistochemistry processing and 3D reconstruction is outlined in [Rojo et al., 2016]. In the present study we measured the angle between two sibling segments originating from a bifurcation of the basal dendritic trees (Figure 1A). Given a bifurcation point O with coordinates (x_0, y_0, z_0) and two points $A = (x_1, y_1, z_1)$ and $B = (x_2, y_2, z_2)$ defining the end points of the segments growing from the bifurcation, the angle ϕ between the vectors OA and OB is given by

$$\phi = arccos\left(\frac{OA \cdot OB}{\|OA\|\|OB\|}\right),$$

where $\cdot$ represents the scalar product of the vectors and $|u|$ is the magnitude of the vector u. The above angles were grouped based on the number of bifurcations that take place in the path that starts at the soma and ends at the angle, meaning that the first bifurcation that takes place in a dendritic arbor would be "Order 1" (denoted by O1), the next possible bifurcations would be "Order 2" (O2), etc. Branch order angles greater than O5 were not included in the analysis due to the relatively low number (Supplementary Table S1).

7.2.2.1 Directional statistics

For this study, we use the vM and JP distributions to model the data. For graphical representation we use the rose diagrams, circular histograms and circular boxplots. See Chapter 2 for further details.

Statistical tests

We used the following statistical tests.

Goodness-of-fit

In order to test the goodness-of-fit to a vM distribution, we used the Watson U^2 test adaptation for the vM distribution [Lockhart and Stephens, 1985] at a significance level of $\alpha = 0.05$ (Supplementary Tables S2-S3). In the case of the JP distribution, we tested the goodness-of-fit using four tests: Rayleigh test [Watson and Williams, 1956], Kuiper test and Rao spacing test [Batschelet, 1981; Upton and Fingleton, 1989; Mardia and Jupp, 2009] and

[2]MicroBrightField url: http://www.mbfbioscience.com/neurolucida

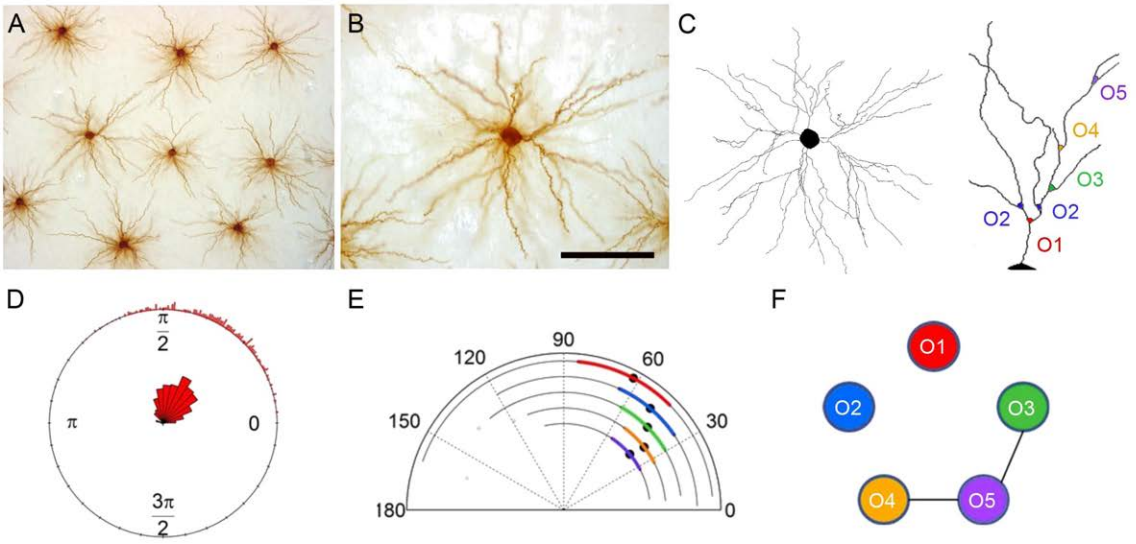

Figure 7.1: (A) Low-power photomicrograph showing injected neurons in layers III from the S1HL region of P14 rats, as seen in the plane of section parallel to the cortical surface. (B) Higher magnification photomicrograph showing an example of a pyramidal cell basal dendritic arbor. (C) Schematic drawing of the basal arbor of the pyramidal neuron shown in B. Angles of different branch orders (shown on the right in different colors)were measured between sibling segments. (D) Example of a rose diagram overlapped with a circular histogram of the distribution of branching angles (in degrees) of the same branch order 1 in layer II. (E) Circular boxplot of the angles showing the summary statistics of a dataset as arcs inside a semicircle. The black dot is the median direction, the colored lines are the boxes (from the lower quartile (Q1) to the upper quartile (Q3)), the black lines are the whiskers that depend on the interquartile range (Q3-Q1) and the concentration parameter (κ) of the distribution, and the colored dots are the outliers that do not belong to the box and whiskers interval. The respective graphs correspond to the comparison between different branch order angles in layer II. (F) Test-based diagram illustrating the pairwise comparisons of the mean angles from datasets shown in D. Two nodes (each node is a dataset) between which there is no statistically significant difference are connected, meaning that the null hypothesis of the Watson nonparametric test cannot be rejected. Scale bar = 200 μm for A; 90 μm for B.

Watson U^2 test [Watson, 1961]. Results for the Watson U^2 test are shown in Supplementary Table S4. We also performed these tests at a significance level of $\alpha = 0.05$.

Comparing the mean direction between datasets

We were also looking for differences between the datasets of angles. Therefore we performed tests to compare the mean directions. In order to compare mean directions between several datasets that fit the vM distribution, we used the Watson-Williams test [Watson and Williams, 1956]. We used the Watson nonparametric test for pairwise comparisons [Watson and Statisticien, 1983]. For datasets that fit the JP distribution, we used the Watson nonparametric test [Watson and Statisticien, 1983] for both comparisons of mean directions between several datasets and pairwise comparisons. We used a significance level of $\alpha = 0.05$ for all the comparison of mean directions tests.

Test-based diagrams

In order to easily visualize the results of the Watson nonparametric pairwise comparison tests, we built a graph (Figure 7.1F) where each node represents a dataset and two nodes that are not statistically significantly different are connected by an edge. This kind of graph has been used before in statistical tests to compare branching angles in cells from different cortical areas [Bielza et al., 2014].

Software

In this Chapter, statistical analysis was performed with R[3] software, and we used circular statistics in the R package [Pewsey et al., 2013].

7.3 Results

We analyzed the branching angles of basal dendrites from 288 pyramidal neurons across layers (II, III, IV, Va, Vb, VI) of the S1HL cortex of P14 rats (Figure 7.1A-C). The images of the 288 reconstructed cells organized by layers are available as supplementary material in [Rojo et al., 2016]. A visual inspection of the rose diagram and the circular histogram (Figure 7.1D and Supplementary Figure 1) revealed that the distribution of the angles were unimodal and symmetric around the mean in all branch orders. The goodness-of-fit test to a vM distribution revealed that this distribution was not good enough to model the branching angles of the same order (Supplementary Table S2), where 14 out of 30 cases were rejected. We searched further for another distribution and found that the JP distribution was appropriate for modelling these angles (Supplementary Table S4), where only two out of 30 cases were rejected. There is a visually appreciable fitting improvement of the JP distribution over the vM distribution (Supplementary Figure 1). The distribution of the angles was further analyzed using the maximum tree order. In this case, the distribution was again found to be unimodal and symmetric around the mean (Supplementary Figure 7). The goodness-of-fit

[3]R url: https://www.r-project.org

test to a vM distribution revealed that this distribution was appropriate for modelling angles of same maximum tree order (Supplementary Table S3).

Angles of different branch order

We used circular boxplots to compare angles of a different branch order in different layers. We observed that the angles tend to decrease as the branch order increases in every layer (Figure 7.1E and Supplementary Figure 2). We also found that the CIQR is the widest at O1 and subsequent orders get narrower. The results of the statistical tests (Supplementary Tables S5-S6) are illustrated in the test-based diagrams (Figure 7.1F and Supplementary Figure 2).Statistically significant differences were found for the angles in the first orders, but angles for higher orders were not significantly different.

Angles of different branch order originating from dendritic trees of similar complexity

We compared angles of different branch orders within dendritic trees that were grouped by the maximum tree order of their arbors. We compared the angles from dendritic trees of the same complexity. Regarding the boxplot (Figure 7.2) and the statistical test results (Supplementary Tables S7-S8) that are illustrated in the test-based diagrams, the analysis revealed even more clearly what we observed without grouping by maximum tree order: there are statistically significant differences between the angles of first orders and there are no significant differences in higher orders. Therefore, by grouping by maximum tree order, we were able to conclude that the branching angles of lower orders are wider than the branching angles of higher orders.

Angles of the same branch order originating from dendritic trees of different complexity

We compared the bifurcation angles of the same branch order that belong to trees of varying complexity. We observed in the boxplots (Figure 7.3) that angles are wider for more complex arbors. This behavior is similar in all cortical layers. However, there is no statistically significant difference (Supplementary Tables S9-S10) between angles of the same branch order that belong to arbors with a maximum tree order greater than three, and the maximum tree order is equal to three in most cases. Additionally, we compared the final branching angles from trees of varying complexity, which, from the boxplots (Supplementary Figure 5), we found to be very similar in all cases. The statistical tests (Supplementary Tables S15-S16) reveal that there are no differences between the final angles of different branch order, and, as we observed graphically in the test-based diagrams, this behavior is the same for every layer.

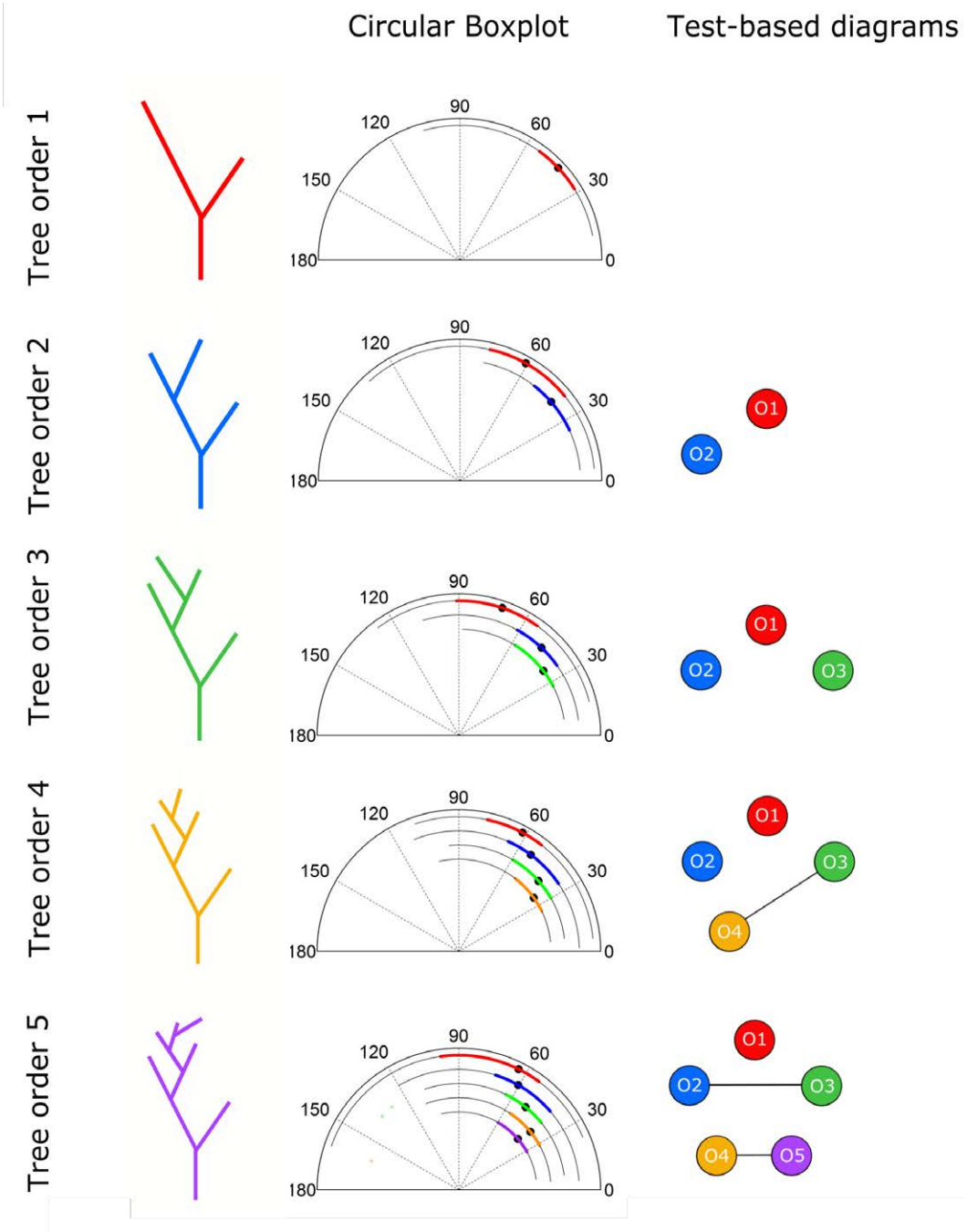

Figure 7.2: Left column: Diagram showing different dendritic arbors of varying complexity (different dendritic trees were grouped according to their maximum branch order). Therefore arbors with only the first bifurcation (O1) would be denoted as "T1" arbors, arbors with a maximum branching order equal to 2 as "T2", etc. Middle column: Circular boxplots showing comparisons of angles of different branch orders from dendritic trees of same maximum tree order from layer II. Right column: The test-based diagrams corresponding to the pairwise statistical test results from Supplementary Tables S7-S8 are illustrated next to each graph. See Supplementary Figure 3 for the remaining layers.

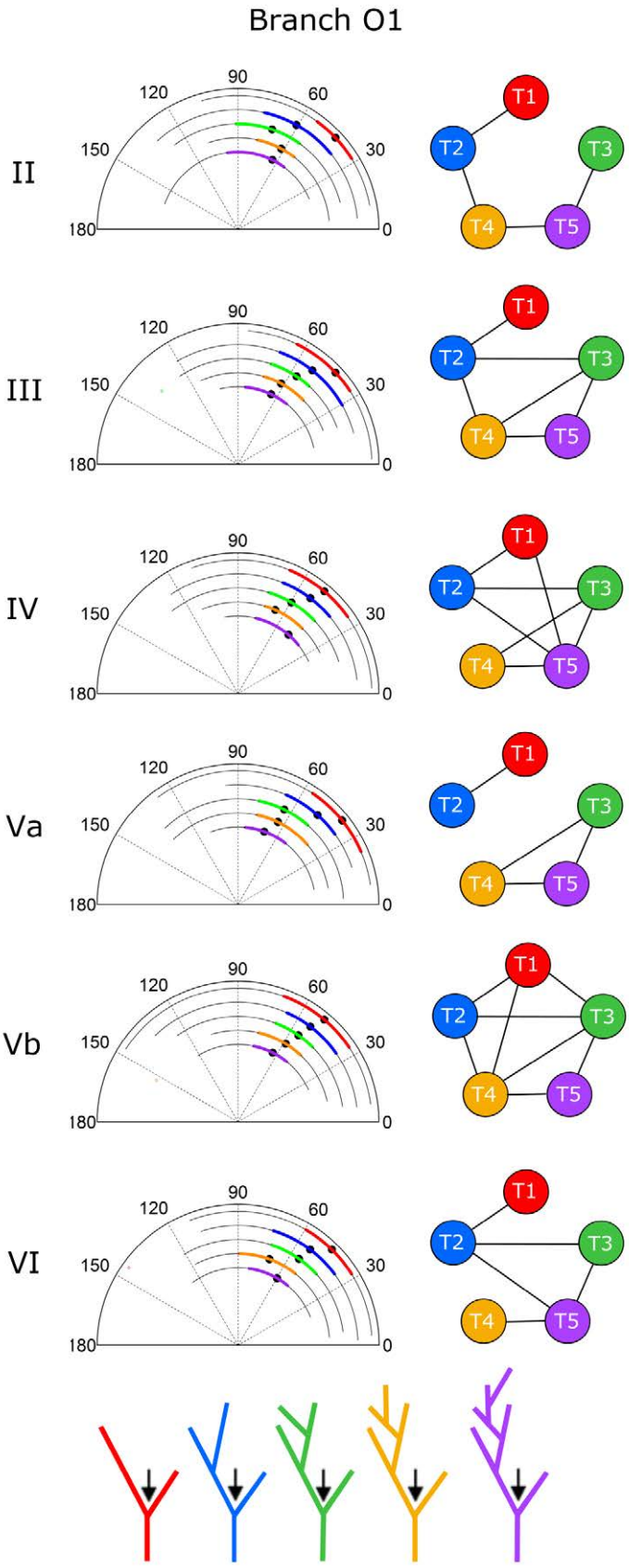

Figure 7.3: Circular boxplots showing comparisons of angles of branch order 1 from dendritic trees of different maximum tree order. The test-based diagrams corresponding to the pairwise statistical test results from Supplementary Tables S9-S10 are illustrated next to each graph. See Supplementary Figure 4 for the remaining branch orders.

Angles of different cortical layers

Finally, we compared angles between layers (II, III, IV, Va, Vb, VI). As shown in Figure 7.4, angles of the same branch order are similar between layers. Nevertheless, we found that the closer the layer in which the neuron is located is to the pia, the less concentrated the distribution of the angles. The statistical test results (Supplementary Tables S11-S12) illustrated in the test-based diagrams (Figure 7.4) showed that there were no statistically significant differences between the angles of the same order from different layers, with the exception of layer II at O1, which did exhibit statistically significant differences. Furthermore, O1 branching angles from layer II are wider than O1 branching angles from the other layers. We also analyzed the differences between branching angles from different layers grouped by trees of similar complexity. An example of the boxplots (Supplementary Figure 6) showed that the distribution of the angles is again less concentrated the closer the layer is to the pia. Similarly, statistical test results (Supplementary Tables S13-S14) showed that there were no statistically significant differences between the branching angles from different layers of the same order in arbors of the same complexity (as also illustrated in the test-based diagram).

7.4 Discussion

The main findings of this study are three: 1) the first bifurcation of a particular basal tree is the widest, and subsequent bifurcations become progressively narrower in all cortical layers; 2) the final bifurcation angle of a dendritic tree is similar regardless of its complexity; and 3) angles of the same branch order are similar to each other in the different cortical layers. We used circular distributions to model the branching angles in 3D reconstructed basal arbors. Previous studies showed that the vM distribution seems to be suitable for modelling the angles generated from dendritic arbor bifurcations in neurons from different cortical areas [Bielza et al., 2014]. Here we reveal that the vM distribution is also suitable for modelling angles in neurons from different layers when grouped according to their maximum tree order, whereas angles grouped just by branch order fit the Jones-Pewsey distribution (a generic circular distribution of which the vM distribution is one instance). Importantly, the results of this and a previous study regarding the geometry of pyramidal cell basal arbors in different cortical areas of adult mice [Bielza et al., 2014] are similar: the first bifurcation of a particular basal tree is the widest and subsequent bifurcations become progressively narrower in both studies. This suggests that the first orders (1 and 2) determine the space that the growing dendritic tree is to fill. In addition, the final bifurcation of a particular tree is rather similar, regardless of the maximum tree order of the arbor. Furthermore, they found, in mice, that 90% of these angles were within a range of $20-97^\circ$ (per cortical area, mean angles ranged from $59-687^\circ$ and concentrations ranged from $5-87^\circ$). These are similar values to the results of this study (angles ranged from $10-1047^\circ$ per cortical layer, mean angles ranged from $41.82-64.177^\circ$ and concentrations ranged from $4.71-9.62$). We should stress that these rules were observed regardless of the differences in the size and complexity of the basal dendritic arbors of these cells between the cortical areas of the mice [Bielza et al., 2014] or between the cortical layers

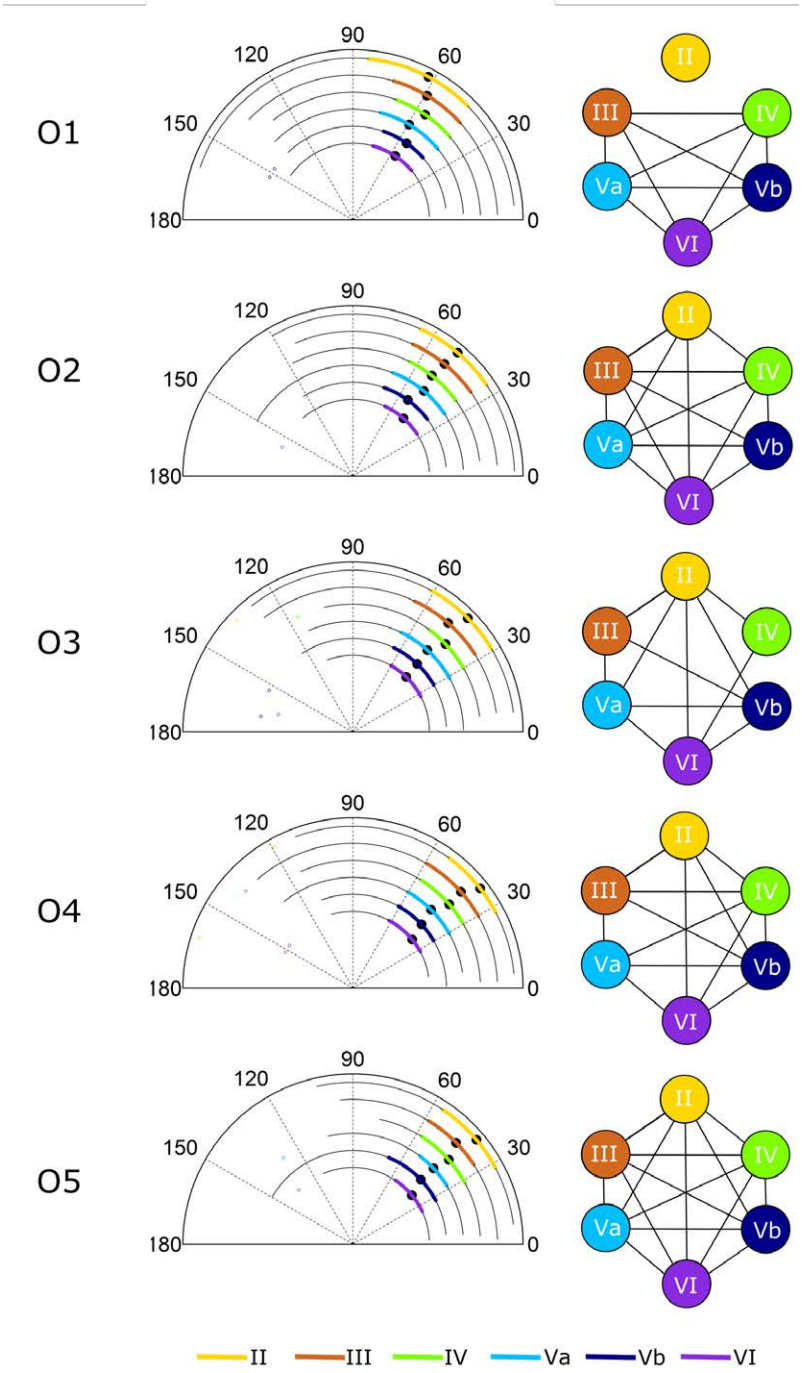

Figure 7.4: Circular boxplots showing comparisons of angles of same branch order in different layers. The test-based diagrams corresponding to the pairwise statistical test results from Supplementary Tables S11-S12 are illustrated next to each graph.

of the rats [Rojo et al., 2016]. Thus, these rules seem to be a general organizational principle in the design of pyramidal cell architecture, despite the different functional specializations of cortical layers and areas and species. In the mouse cerebral cortex, however, it was observed that the mean final branch order angle was remarkably different in the seven examined cortical regions [Bielza et al., 2014]. In general, cortical regions with larger dendritic trees had smaller final bifurcation angles. However, no significant differences were found between the branch order angles of pyramidal cells across layers of the juvenile rat somatosensory cortex despite the systematic variation in the basal dendritic pattern [Rojo et al., 2016]. Briefly, cells became larger and progressively more complex in their branching structure from superficial to deeper layers, except for those in layer IV, which were the simplest cells. Taken together, these results suggest that the final branch order angle may constitute an area-specific feature. Further studies of the different dendritic compartments (e.g. apical arbor), cortical regions and species would need to be performed to make such a generalization. In addition, since we examined juvenile rats, it would be interesting to analyze if branching angle structure in the adult rat cortex remains the same as in the juvenile rats in order to make species comparisons. Clearly, it is of critical importance to determine these rules since general principles of cortical synaptic connections also exist. Therefore, the integration of the morphological rules of pyramidal cells with the principles of their synaptic connection is fundamental in order to gain a better understanding of the design of cortical circuits. For instance, most excitatory, glutamatergic synapses on pyramidal neurons are established with their dendritic spines, whereas most inhibitory GABAergic synapses are established mainly in the dendritic shafts, but the vast majority of synapses are established on the dendritic spines [reviewed in DeFelipe and Fariñas, 1992], the length of which is typically $< 2\mu$m, (e.g., [Ballesteros-Yáñez et al., 2006; Benavides-Piccione et al., 2012]). Therefore, differences in the complexity, dendritic length and dendritic spine density of the dendritic tree between layers reflect differences in the total number of excitatory and inhibitory synapses in the pyramidal neurons. However, the fact that no significant differences were found between the branch order angles of pyramidal cells across layers suggests that there is some predictability in the synaptic connections of pyramidal cells in all cortical layers that is independent of the total number of synaptic inputs. Thus, the variations in pyramidal cell structure indicate that the cortical circuits in which these cells participate are likely to be characterized by different functional capabilities (integration of excitatory and inhibitory synapses). However, we do not know whether the branch angles have a significant direct impact on signal processing per se. Computational simulations performed by Ferrante et al. [Ferrante et al., 2013] have shown that minor changes in dendritic branch-point morphology of CA1 apical trees of pyramidal cells can lead to major modifications in the integrative properties of oblique dendrites. In this regard, further computational modelling studies could also contribute towards attempts to predict the biophysical consequences of varying branch angles of the basal dendrites from the first (the wider) to the subsequent bifurcations which become progressively narrower. A further point to note is the fact that the structure between the branch order angles of pyramidal cells is unchanged across layers, which supports the idea that the factors that

intrinsically regulate dendritic branching development are probably related to the rules that determine the general connectivity of the pyramidal cell. More specifically, our results seem to indicate the existence of spatial synaptic connectivity rules of pyramidal neurons which are constrained by the relatively narrow value windows of the bifurcation angles. Finally, the computational attributes of pyramidal cells do not only depend on their basal dendritic arbors, but also on the structure of their apical dendrites. Thus, we are planning to address some of these questions by analyzing the apical arbor in the near future. Additional studies in other species/cortical areas and ages are necessary to further elucidate the generally applicable and specific rules governing the geometry of cortical pyramidal cells.

Chapter 8

Bayesian network-based circular classifiers for dendritic branching angles of pyramidal cells

8.1 Introduction

As stated in Chapters 4 and 7, the dendritic bifurcation angles produced by the branch of the dendrites are an important part of the geometry of pyramidal cell arbors. Understanding and modelling them is challenging but crucial for advances in neuroscience to replicate brain functioning and structure.

Predicting which layer a neuron belongs to is an important task to help understand any neural circuit, and it represents part of the picture regarding the identification and characterization of all its components. To the best of our knowledge, there is no any supervised classification model that predicts the layer using circular predictive variables. Thus, in this Chapter we use the real neuromorphological dataset obtained from juvenile rat somatosensory cortex cells introduced in Chapter 7, where we measure the bifurcation angles of the dendritic basal arbors. Here, we use the classification models for circular data presented in Chapter 5 to predict which layer a given neuron belongs to, i.e., layers II, III, IV, Va, Vb or VI.

Chapter outline

Section 8.2 addresses the real-world neuromorphology data problem using the wrapped Cauchy classifiers. In Section 8.3 the conclusions are reported.

8.2 Results

In this Section, we apply the GTAN and the classification models presented in Chapter 5 to the dataset used in Chapter 7. This consists of 3027 combinations of dendritic bifurcation

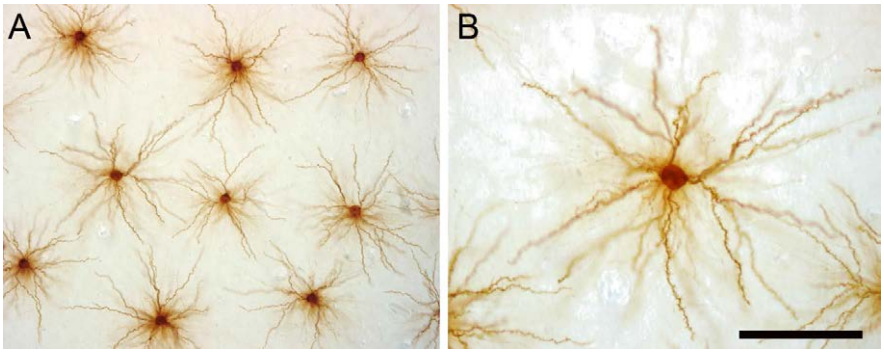

Figure 8.1: (A) Low-power photomicrograph showing injected neurons in layers III from the S1HL region of P14 rats, as seen in the plane of section parallel to the cortical surface. (B) Higher magnification photomicrograph showing an example of a pyramidal cell basal dendritic arbor. Scale bar (in B) = 200μm in A; 90μm in B. Adapted from [Leguey et al., 2016b].

Bifurcation Order	Variable	Number of angles	$\hat{\mu}$ (in radians)	$\hat{\varepsilon}$
1	Θ_1	1607	1.02	0.90
2	Θ_2	2072	0.90	0.91
3	Θ_3	1773	0.82	0.92
4	Θ_4	998	0.78	0.92
5	Θ_5	382	0.77	0.92
6	Θ_6	106	0.81	0.92

Table 8.1: Characteristics of the six different branching orders shown in Fig. 8.2.

angles coming from the basal arbors of 288 3D pyramidal neurons in layers II, III, IV, Va, Vb and VI (48 neurons per layer) of the P14 rat S1HL neocortex, published in [Leguey et al., 2016b] (Fig. 8.1).

We model the bifurcation angles produced by the splitting of the dendritic segments of basal dendritic trees. Following the notation used in Chapter 7, Θ_1 will correspond to the first bifurcation angle (Order 1) generated for the first split of the dendritic segments starting from the soma. The second angle generated by the next consecutive splits will be represented as variable Θ_2 (Order 2), etc. (Fig. 8.2). Angles of orders higher than six which were relatively scarce were not included in the model. Therefore, following the notation stated in Chapter 5, the vector of circular predictor features is $\boldsymbol{\Theta} = (\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6)$ and the discrete class variable C is de layer, which takes values in the set $\Lambda(C)$ =(II, III, IV, Va, Vb, VI). For each set of angles of the same order, a wC distribution was fitted (Table 8.1). We performed a goodness-of-fit test by transformation on the circle of the variables into circular uniform variables via $2\pi F(\Theta_1), .., 2\pi F(\Theta_6)$, where F is the cumulative distribution function, and applied Kuiper's test [Kuiper, 1960] for circular uniformity with a significance level of $\alpha = 0.05$.

Note that in Table 8.1 the circular mean tends to decrease as the order increases. A neuroscientific explanation for this behaviour relates to the fact that it is the first bifurcation orders that determine the volume of space to be filled by the dendritic trees [Leguey et al.,

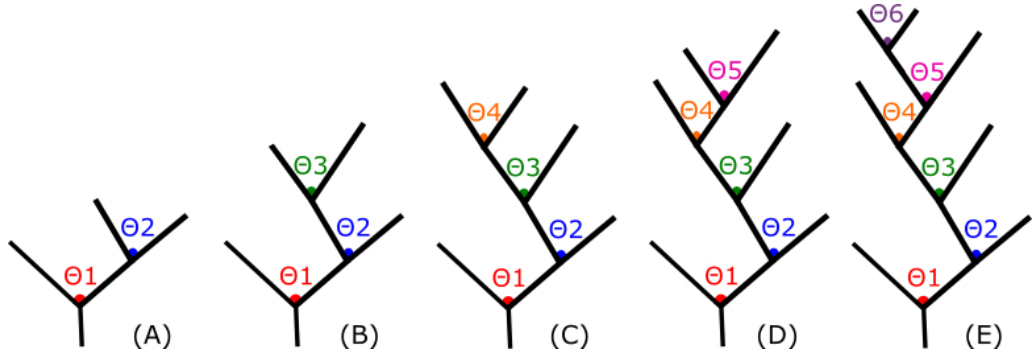

Figure 8.2: Angles of different branch orders (from 1 to 6) measured between sibling segments in a dendritic arbor. The dendritic arbor has a maximum branching order of (A) 2 (B) 3 (C) 4 (D) 5 (E) 6.

		Classifiers				
		wCNB	wCsNB	wCsmNB	wCTAN	wCTAN
Max. bifurc. angle	2	0.182 ± 0.034	**0.184 ± 0.055**	0.182 ± 0.039	0.182 ± 0.034	0.158 ± 0.038
	3	0.191 ± 0.035	**0.219 ± 0.035**	0.203 ± 0.057	0.205 ± 0.057	0.113 ± 0.030
	4	0.222 ± 0.016	0.224 ± 0.034	**0.239 ± 0.057**	0.222 ± 0.046	0.212 ± 0.081
	5	0.196 ± 0.091	0.235 ± 0.127	**0.239 ± 0.063**	0.189 ± 0.055	0.128 ± 0.131
	6	0.220 ± 0.113	0.270 ± 0.094	**0.290 ± 0.137**	0.240 ± 0.126	0.047 ± 0.069

Table 8.2: Mean ± standard deviation of layers II, III, IV, Va, Vb and VI classification accuracy results of the battery of classifiers for each type of classifier applied over the dataset of dendritic bifurcation angles coming from the basal arbors of 288 3D pyramidal neurons of P14 rat S1HL neocortex.

2016b]. This regulates the dendritic branching development rules that seem to determine the synaptic connectivity of pyramidal neurons. We also observe that the concentration values are high (around 0.91) and quite similar in every bifurcation order. This fact demonstrates that the dendritic structure (in terms of bifurcation angles) is determined by the location parameter.

Since not all dendritic arbors present angles of all orders, one classifier for the whole dataset is not suitable. Therefore, for each classification model proposed in this paper, we created a battery of five classifiers depending on the maximum bifurcation order of the arbor, when this is higher than 1 (Fig. 8.3). Before predicting class c^*, we have to check the maximum bifurcation order of the instance to be classified. For the wCTAN and GTAN structures (which require a root node in addition to the class node) we select as root node Θ_2 for every classifier of the battery. We performed 10 fold cross-validation procedures in order to obtain the mean classification accuracy values for each classifier and maximum bifurcation order (Table 8.2).

We observe in Table 8.2 that the wCsNB classifier leads to the best results for arbors with a maximum branching order of Θ_2 and Θ_3. Furthermore, for arbors with a maximum branching order of Θ_4, Θ_5 or Θ_6, the wCsmNB seems to perform best in terms of classifica-

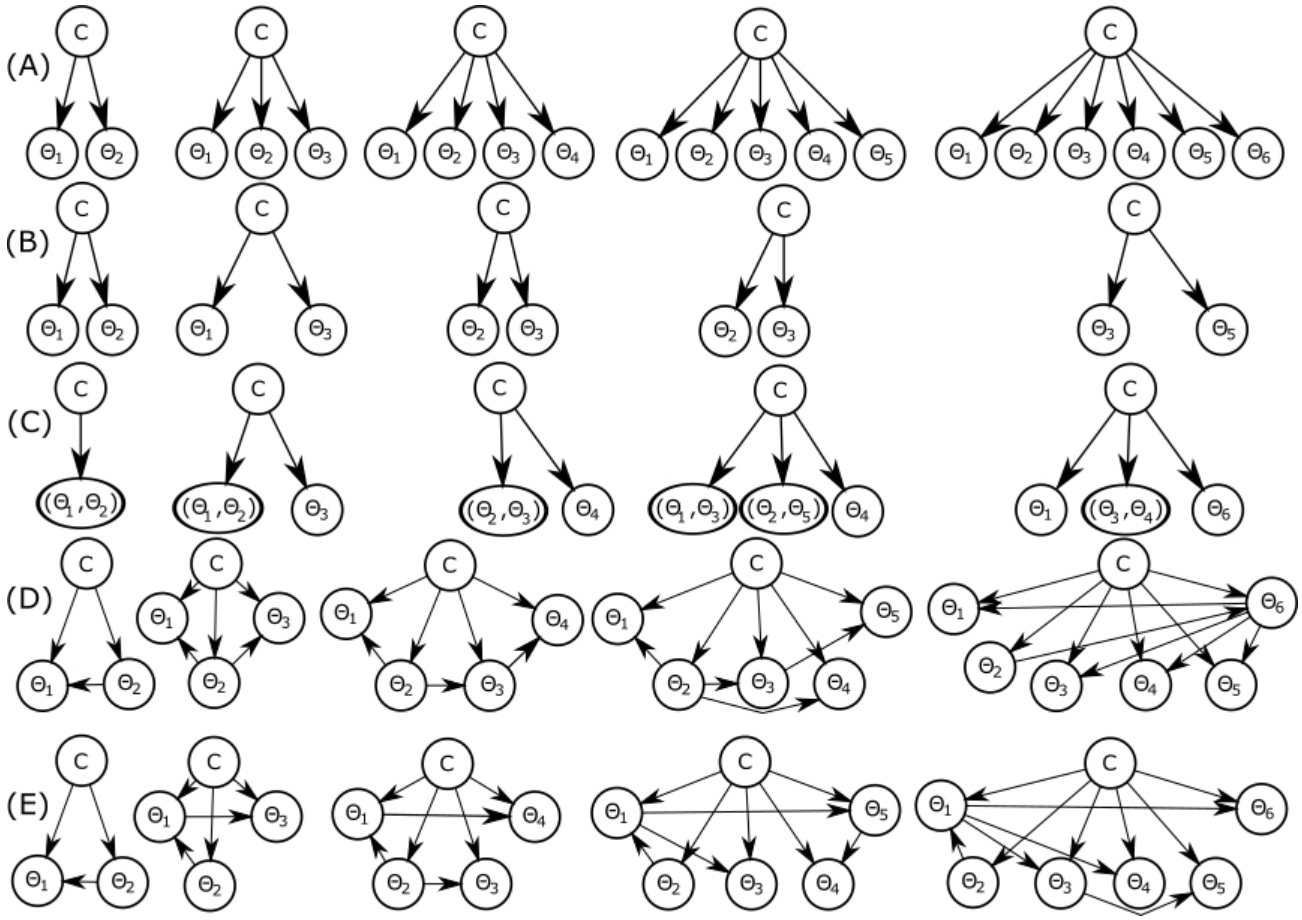

Figure 8.3: Bayesian network classifier structures associated with the battery of classifiers depending on the maximum bifurcation order, for each type of classification algorithm: (A) wCNB, (B) wCsNB, (C) wCsmNB, (D) wCTAN, (E) GTAN.

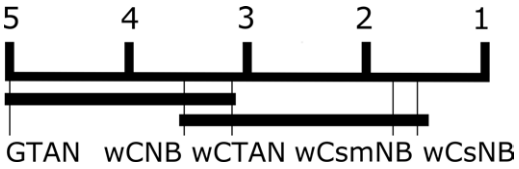

Figure 8.4: Demšar diagram for the comparison of wCNB, wCsNB, wCsmNB, wCTAN and GTAN classification models using Friedman test and Nemenyi post-hoc test.

tion accuracy. The wCTAN and wCNB classifiers also report acceptable values in comparison with the highest ones for each maximum bifurcation arbor, although the wCsNB or wCsmNB classification models always perform better for this neuronal dataset. Comparing the accuracy results with the random label assignation (i.e., $1/6 = 0.16$), we observe that all of these results are over 0.16. In addition, for every case, the GTAN classifier exhibits the lowest accuracy values, below 0.16 except for Θ_4. This classifier was especially inaccurate for arbors that had maximum branching order of 6; the mean accuracy value was 0.047 for such cases.

We applied the Friedman non-parametric test (see Section 5.4 for detailed information) to detect statistically significant differences in the results provided by our algorithms. Since the null hypothesis was rejected (p-value = 0.004), we used Nemenyi post-hoc test (further information in Section 5.4) to determine which pairwise of algorithms was the cause of the Friedman test rejection. In Figure 8.4, the statistically significant differences between our classifiers are represented as a Demšar diagram (see Section 5.4 for further information). We noted that there are no statistically significant differences between our classification algorithms except for two cases; between the wCTAN and wCsNB and between GTAN and the wCsmNB.

Therefore, we can conclude that (i) apart from the difficulty identifying the layer a case belongs to, it seems reasonable to use any of our four proposed circular classifiers for this neuronal dataset, since there are no any statistically significant differences between them and (ii) GTAN is never recommended.

8.3 Conclusions

In this chapter, we evaluated a battery of classifiers using a real-world neuroscience dataset for each of the classifiers presented in Chapter 5, in order to predict the layer that an instance belongs to. Results revealed that all our four classification models are suitable. Performing Friedman test and its corresponding Nemenyi post-hoc test after rejection, we realised that there are no any statistically significant differences between wCNB, wCsNB, wCsmNB and wCTAN for this dataset. Wrapped Cauchy classifiers always outperformed their linear (Gaussian) counterparts.

Regarding the limitations of our model, stated in Chapter 5, it seems interesting to repeat this experiment and compare the results using an extension of these models, able to consider the (possible) multivariate relationships between the bifurcation angles from different orders.

Part V

CONCLUSIONS

Chapter 9

Conclusions and future work

This chapter summarizes the most important contributions and describes some future work and open issues. The chapter also shows a list of publications and submissions produced in this research.

Chapter outline

Section 9.1 summarizes the main contributions and conclusions reported in this dissertation. Section 9.2 includes the list of publications and current submissions produced during this research. Finally, in Section 9.3 the future work and open issues are discussed.

9.1 Summary of contributions

The contributions are organized in two parts:

- Part III includes our contribution to Bayesian networks and directional statistics. In Chapter 5 we present a set of supervised classification models capable of dealing with circular wrapped Cauchy predictive variables. These are the wrapped Cauchy naive Bayes, the wrapped Cauchy selective naive Bayes, the wrapped Cauchy semi-naive Bayes and the wrapped Cauchy tree-augmented naive Bayes classifiers. The chapter details the four classification models and describes the experimental process performed to evaluate their behaviour. We find that given circular datasets these wrapped Cauchy classifiers perform classification accurately. We also demonstrate that these circular classifiers outperform linear classifiers for datasets of circular nature that follow wrapped Cauchy distributions. Based on the conducted statistical tests, the wrapped Cauchy naive Bayes classifier, wrapped Cauchy semi-naive Bayes classifier and wrapped Cauchy tree-augmented naive Bayes classifier outperform the results obtained for the wrapped Cauchy selective naive Bayes with no statistical differences among them.

  In Chapter 6 we go one step further to combine circular variables with linear variables. In this chapter, we present the circular-linear mutual information measure and extend

the definition of the circular mutual information measure. We show that these measures can be expressed in a simple and closed form. The chapter describes the experimental process, where we illustrate an application of these measures in a Bayesian network model. This model is tree-structured and is capable to capture the dependence between any possible pair of linear and circular variables. In this chapter, we also apply this circular-linear Bayesian network model to the study of the relationship between several meteorological variables with circular and linear nature recorded in Europe. We demonstrate that our proposal outperforms other models, which assume that all variables are Gaussian or discrete.

- Part IV includes our contribution to neuroscience. In particular, this part is focused on the basal dendritic structure of pyramidal cells (i.e., neuron morphology). The studies are conduced over a set of 288 pyramidal neurons from six different layers of the 14-day-old rat hind limb somatosensory neocortex. In Chapter 7 we use circular distributions to model the branching angles in 3D reconstructed basal arbors. We show that the von Mises distribution is suitable to model angles from the dendritic bifurcations in neurons from different layers when grouped according to their maximum arbor order. Nevertheless, when the branching angles are grouped by bifurcation order the von Mises distribution is not suitable to model them, whereas the Jones-Pewsey distribution is. This chapter also shows, that concurring with previous studies in mice [Bielza et al., 2014], the first bifurcation of a particular basal arbor is the widest and subsequent bifurcations become progressively narrower (i.e., the first orders (1 and 2) determine the space that the growing dendritic arbor is to fill). The results reported in this study also suggest that cells become larger and progressively more complex in their branching structure from superficial to deeper layers, except for those in layer IV, which were the simplest cells. Furthermore, our results seem to indicate the existence of spatial synaptic connectivity rules of pyramidal neurons that are constrained by the relatively narrow value windows of the bifurcation angles.

  In Chapter 8 we apply the classification models presented in Chapter 5 over the set of dendritic bifurcation angles studied in Chapter 7. In this chapter, we model the branching angles of the basal dendrites from pyramidal neurons using the wrapped Cauchy distribution. We apply the wrapped Cauchy classifiers to predict which layer a neuron belongs to, helping to understand any neural circuit. Owing to data nature, we perform classification with a battery of classifiers grouping the branching angles by its maximum dendritic arbor order. The statistical tests results performed for this study show that all the four classification models outperform their linear counterparts and are suitable to identify the layer that a neuron belongs to, based on its basal dendritic bifurcation angles.

- Following the above conclusions, we achieve the preliminary objectives of this dissertation listed in Chapter 1. Therefore, we verify the original hypotheses that motivate this work:

– We demonstrate that Bayesian network models can be improved using directional statistics techniques to deal with circular variables.

– We develop a model capable to predict the cerebral cortex layer of a neuron given the information of its basal dendritic branching angles.

9.2 List of publications

The conducted research for this dissertation has produced to the following dissemination results:

Peer-reviewed JCR journals

- C. Rojo, I. Leguey, A. Kastanauskaite, C. Bielza, P. Larrañaga, J. DeFelipe, and R. Benavides-Piccione. Laminar differences in dendritic structure of pyramidal neurons in juvenile rat somatosensory cortex. *Cerebral Cortex*, 26(6):2811-2822, 2016.

- I. Leguey, C. Bielza, P. Larrañaga, A. Kastanauskaite, C. Rojo, R. Benavides-Piccione and J. DeFelipe. Dendritic branching angles of pyramidal cells across layers of the juvenile rat somatosensory cortex. *Journal of Comparative Neurology*, 524(13):2567-2576, 2016.

- P. Fernandez-Gonzalez, R. Benavides-Piccione, I. Leguey, C. Bielza, P. Larrañaga, and J. DeFelipe. Dendritic branching angles of pyramidal neurons of the human cerebral cortex. *Brain Structure and Function*, 222(4):1847-1859, 2017.

- I. Leguey, C. Bielza and P. Larrañaga. Circular Bayesian classifiers using wrapped Cauchy distributions. *Submitted*, 2017.

- I. Leguey, P. Larrañaga, C. Bielza and S. Kato. A circular-linear dependence measure under Johnson–Wehrly distributions and its application in Bayesian networks. *Submitted*, 2017.

- I. Leguey, R. Benavides-Piccione, C. Rojo, P. Larrañaga, C. Bielza and J. DeFelipe. Patterns of dendritic basal field orientation of pyramidal neurons in the rat somatosensory cortex. *Submitted*, 2018.

Communications

- I. Leguey, C. Bielza and P. Larrañaga. Tree-structured Bayesian networks for wrapped Cauchy directional distributions. In: *Advances in Artificial Intelligence, Proceedings of the 17th Conference of the Spanish Association for Artificial Intelligence, CAEPIA*, volume 9868 of *Lecture Notes in Artificial Intelligence*, pages 207-216, Springer, 2016.

- I. Leguey, S.Kato, C. Bielza and P. Larrañaga. Hybrid mutual information. In: *Advances in Directional Statistics, ADISTA Workshop*, Rome, 2017. *2nd best poster award.*

9.3 Future work

This section summarizes the future work and open issues of the research conducted in this dissertation. Detailed discussions can be found in the specific section of each chapter.

In this dissertation, as part of the contributions to Bayesian networks and directional statistics, we propose four supervised classification wrapped Cauchy classifiers in Chapter 5. The models presented are limited to no more than bivariate relationships. Therefore to extend these circular classification models to other more sophisticated, like k-dependence wrapped Cauchy classifiers, capable to represent and take into account multivariate relationships between circular variables is an open issue.

We also intend to extend our work presented in Chapter 6 and adapt the proposed circular-linear graphical model to supervised classification models. Furthermore, the constraint dimension of the model to one parent is another interesting task to develop, extending this model to a more general case that allows more than one parent per node. This is a difficult task owing to the non-closed nature of the circular families that are known to date. Therefore, the development of a family of multivariate circular distributions whose marginals and conditionals belong to the same family, is also a challenge.

Regarding to the applications in neuroscience developed in this dissertation, in Chapter 7 we study and model the branching angles of the basal dendritic arbors of pyramidal neurons. We find that the final branch order angle may constitute an area-specific feature. Nevertheless, further studies of the different dendritic compartments (e.g., apical arbor), cortical regions, and species would need to be performed to make such a generalization. In addition, for this experiment we examined juvenile 14-days-old rats, therefore it would be interesting to analyze if branching angle structure in the adult rat cortex remains the same as in the juvenile rats in order to make species comparisons.

In Chapter 8 we classify the pyramidal neurons regarding to their branching angles of the basal dendritic arbors. Since the models used for this classification are those presented in Chapter 5, it would be interesting to extend this analysis using circular classification models capable to deal with multivariate relationship between the variables.

The basal dendritic arbors play an important role on the pyramidal neurons characterization. Additional studies to make further on this issue are critical. For example, we intend to study the orientation of the basal dendrites of pyramidal neurons in order to find whether nearby neurons present common basal dendritic growing orientation.

Furthermore, the computational attributes of pyramidal cells depend not only on their basal dendritic arbors, but also on the structure of their apical dendrites. Thus, we intend to address some of the open issues in Chapters 7 and 8 by analyzing the apical arbor. Additional studies in other species/cortical areas and ages are necessary to further elucidate the general

and specific rules governing the geometry of cortical pyramidal cells.

Part VI

APPENDICES

Appendix A

Circular-linear dependence measures under Wehrley–Johnson distributions

A.1 Proof of Theorem 6.1

The conditional density function of Θ, given $X = x$, and that of X, given $\Theta = \theta$, can be expressed as

$$f(\theta|x) = \frac{f(\theta, x)}{f_X(x)}$$

and

$$f(x|\theta) = \frac{f(\theta, x)}{f_\Theta(\theta)},$$

respectively. Changing the variable $U = 2\pi F_\Theta(\Theta)$ in $f(\theta|x)$, we obtain

$$f(u|x) = f(\theta|x)\left|\frac{\partial\theta}{\partial u}\right| = g(u - 2\pi q F_X(x)),$$

where $q \in \{-1, 1\}$. Similarly, changing the variable $V = 2\pi F_X(X)$ in $f(x|\theta)$ leads to

$$f(v|\theta) = f(x|\theta)\left|\frac{\partial x}{\partial v}\right| = g(2\pi F_\Theta(\theta) - qv).$$

Let $g(.)$ be the wrapped Cauchy density function given by Equation (2.4), with location parameter μ_g and concentration parameter ε_g, as defined in Kato [Kato, 2009]. Then, the following hold:

$$U|X = x \sim wC(2\pi q F_X(x) + \mu_g, \varepsilon_g)$$

and

$$V|\Theta = \theta \sim wC(q(2\pi F_\Theta(\theta) - \mu_g), \varepsilon_g). \tag{A.1}$$

Consider the case where $\Theta \sim wC(\mu_\theta, \varepsilon_\theta)$ and $X \sim N(\iota_x, \sigma_x^2)$. Without loss of generality, the origin of the cumulative distribution function of Θ is assumed to be zero (i.e. $F_\Theta(\theta) = \int_0^\theta f_\Theta(t)dt$). In this case, the conditional of X, given $\Theta = \theta$, does not follow any well-known distribution. However, Equation (A.1) implies that

$$2\pi\,\Phi\left(\frac{X-\iota_x}{\sigma_x}\right)|\Theta = \theta \sim wC(q(2\pi F_\Theta(\theta) - \mu_g), \varepsilon_g),$$

where Φ denotes the CDF of the standard Gaussian distribution $N(0,1)$, namely, $\Phi(x) = \int_{-\infty}^{x}\phi(t)dt$, where ϕ is the Gaussian density with $\iota_x = 0$ and $\sigma_x = 1$. Since it is easy to evaluate the CDF of the standard Gaussian distribution numerically, numerical calculations associated with the conditional distribution of X, given $\Theta = \theta$, can be conducted efficiently.

The conditional of Θ, given $X = x$, has a wrapped Cauchy distribution. To see this, we first note that U and Θ have the following relationship:

$$e^{(iU)} = e^{(i\nu)}\frac{e^{(i\Theta)} - \varepsilon_\theta e^{(i\mu_\theta)}}{1 - \varepsilon_\theta e^{(i(\Theta-\mu_\theta))}}$$

or

$$e^{(i\Theta)} = \frac{e^{(i(U-\nu))} + \varepsilon_\theta e^{(i\mu_\theta)}}{1 + \varepsilon_\theta e^{(i(U-\nu-\mu_\theta))}},$$

where $e^{(i\nu)} = (1 - \varepsilon_\theta e^{(-i\mu_\theta)})/(1 - \varepsilon_\theta e^{(i\mu_\theta)})$. McCullagh [McCullagh, 1996] showed that

$$e^{(iU)} \sim C^*(\alpha_1 e^{(i\beta_1)}) \implies \frac{e^{(iU)} + \alpha_2 e^{(i\beta_2)}}{1 + \alpha_2 e^{(i(U-\beta_2))}} \sim C^*\left(\frac{\alpha_1 e^{(i\beta_1)} + \alpha_2 e^{(i\beta_2)}}{1 + \alpha_1\alpha_2\exp\left(i(\beta_1 - \beta_2)\right)}\right),$$

where $0 \le \alpha_1, \alpha_2 < 1$ and $-\pi < \beta_1, \beta_2 \le \pi$. Using this result, we have $e^{(i\Theta)}|X = x \sim C^*(\hat{\phi}_{\theta|x})$ or, in polar-coordinate form, $\Theta|X = x \sim wC(arg(\hat{\phi}_{\theta|x}), |\hat{\phi}_{\theta|x}|)$, where

$$\hat{\phi}_{\theta|x} = \frac{\varepsilon_g e^{(i(2\pi q F_X(x) + \mu_g - \nu))} + \varepsilon_\theta e^{(i\mu_\theta)}}{1 + \varepsilon_g\varepsilon_\theta e^{(i(2\pi q F_X(x) + \mu_g - \mu_\theta - \nu))}}.$$

A.2 Proof of Theorem 6.2

Let $f(\theta, \psi)$ be the joint density function given by Equation (6.3) and defined in Johnson and Wehrly [Wehrly and Johnson, 1980]. Then, the CMI between Θ and Ψ is given by

$$\begin{aligned}&\mathrm{CMI}(\Theta, \Psi)\\ &= \int_0^{2\pi}\int_0^{2\pi} 2\pi\delta(2\pi F_\Theta(\theta) - q2\pi F_\Psi(\psi)) f_\Theta(\theta) f_\Psi(\psi)\log\left\{2\pi\delta(2\pi F_\Theta(\theta) - q2\pi F_\Psi(\psi))\right\} d\psi d\theta.\end{aligned}$$

Changing the variables $U = 2\pi F_\Theta(\Theta)$ and $V = 2\pi F_\Psi(\Psi)$, we have

$$\mathrm{CMI}(\Theta, \Psi) = \frac{1}{2\pi}\int_0^{2\pi}\int_0^{2\pi}\delta(u - qv)\log\left\{2\pi\delta(u - qv)\right\} dudv.$$

It follows from the change of variables $t_1 = u - qv$ and $t_2 = v$ that

$$\mathrm{CMI}(\Theta, \Psi) = \int_0^{2\pi} \delta(t_1) \log \{2\pi\delta(t_1)\}\, dt_1.$$

Let $\delta(.)$ be the wrapped Cauchy density function given by Equation (2.4), with location parameter μ_δ and concentration parameter ε_δ, as defined in Kato [Kato, 2009]. Then,

$$\begin{aligned}\mathrm{CMI}(\Theta, \Psi) &= \int_0^{2\pi} \frac{1}{2\pi} \frac{1-\varepsilon_\delta^2}{1+\varepsilon_\delta^2 - 2\varepsilon_\delta \cos(t_1)} \log\left\{\frac{1-\varepsilon_\delta^2}{1+\varepsilon_\delta^2 - 2\varepsilon_\delta \cos(t_1)}\right\} dt_1 \\ &= -\log(1-\varepsilon_\delta^2),\end{aligned}$$

where the second equality follows from Equation (4.396.16) of Gradshteyn and Ryzhik [Gradshteyn and Ryzhik, 2007].

A.3 Proof of Theorem 6.3

Theorem 6.3 can be proved in a similar manner to Theorem 6.2 by changing the variables $U = 2\pi F_\Theta(\Theta)$ and $V = 2\pi F_X(X)$.

Bibliography

T. Abe and C. Ley. A tractable, parsimonious and flexible model for cylindrical data, with applications. *Econometrics and Statistics*, 4:91–104, 2016.

A. H. Abuzaid, I. B. Mohamed, and A. G. Hussin. Boxplot for circular variables. *Computational Statistics*, 27(3):381–392, 2012.

C. Agostinelli and U. Lund. *Circular: Circular statistics*, 2013. URL http://CRAN.R-project.org/package=circular.

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

A. P. Alivisatos, M. Chun, G. M. Church, R. J. Greenspan, M. L. Roukes, and R. Yuste. The brain activity map project and the challenge of functional connectomics. *Neuron*, 74(6): 970–974, 2012.

A. P. Alivisatos, M. Chun, G. M. Church, K. Deisseroth, J. P. Donoghue, R. J. Greenspan, P. L. McEuen, M. L. Roukes, T. J. Sejnowski, and P. S. Weiss. The brain activity map. *Science*, 339(6125):1284–1285, 2013.

F. R. Bach and M. I. Jordan. Learning graphical models with Mercer kernels. In *Advances in Neural Information Processing Systems 15*, pages 1033–1040, 2003.

I. Ballesteros-Yáñez, R. Benavides-Piccione, G. Elston, R. Yuste, and J. DeFelipe. Density and morphology of dendritic spines in mouse neocortex. *Neuroscience*, 138(2):403–409, 2006.

E. Batschelet. *Circular Statistics in Biology*. Academic Press, 1981.

R. Benavides-Piccione, F. Hamzei-Sichani, I. Ballesteros-Yáñez, J. DeFelipe, and R. Yuste. Dendritic size of pyramidal neurons differs among mouse cortical regions. *Cerebral Cortex*, 16(7):990–1001, 2006.

R. Benavides-Piccione, I. Fernaud-Espinosa, V. Robles, R. Yuste, and J. DeFelipe. Age-based comparison of human dendritic spine structure using complete three-dimensional reconstructions. *Cerebral Cortex*, 23(8):1798–1810, 2012.

C. Bielza and P. Larrañaga. Discrete Bayesian network classifiers: A survey. *ACM Computing Surveys*, 47(1):Article 5, 2014.

C. Bielza, R. Benavides-Piccione, P. López-Cruz, P. Larrañaga, and J. DeFelipe. Branching angles of pyramidal cell dendrites follow common geometrical design principles in different cortical areas. *Scientific Reports*, 4:Article 5909, 2014.

R. Blanco, I. Inza, and P. Larrañaga. Learning Bayesian networks in the space of structures by estimation of distribution algorithms. *International Journal of Inteligent Systems*, 18 (5):205–220, 2003.

R. Blanco, I. Inza, M. Merino, J. Quiroga, and P. Larrañaga. Feature selection in Bayesian classifiers for the prognosis of survival of cirrhotic patients treated with TIPS. *Journal of Biomedical Informatics*, 38(5):376–388, 2005.

R. R. Bouckaert, E. Castillo, and J. Gutiérrez. A modified simulation scheme for inference in Bayesian networks. *International Journal of Approximate Reasoning*, 14(1):55–80, 1996.

K. Bowman and L. Shenton. Methods of moments. *Encyclopedia of Statistical Sciences*, 5: 467–473, 1985.

A. Cano, M. Gémez-Olmedo, and S. Moral. Approximate inference in Bayesian networks using binary probability trees. *International Journal of Approximate Reasoning*, 52(1): 49–62, 2011.

D. Cartwright. The use of directional spectra in studying output of a wave recorder on a moving ship. In *Proceedings of the Conference on Ocean Wave Spectra*, pages 203–218. Prentice-Hall, Inc., 1963.

A. M. Carvalho, T. Roos, A. L. Oliveira, and P. Myllymäki. Discriminative learning of Bayesian networks via factorized conditional log-likelihood. *Journal of Machine Learning Research*, 12:2181–2210, 2011.

J. Cheng and R. Greiner. Learning Bayesian belief network classifiers: Algorithms and system. In *Advances in Artificial Intelligence*, pages 141–151. Springer, 2001.

D. M. Chickering. Learning Bayesian networks is NP-complete. In *Learning from data: Artificial Intelligence and Statistics V*, volume 112, pages 121–130. Springer, 1996.

D. M. Chickering, D. Geiger, and D. Heckerman. Learning Bayesian networks is NP-hard. Technical report, MSR-TR-94-17, Microsoft Research, 1994.

C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

G. F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42(2-3):393–405, 1990.

G. F. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347, 1992.

C. Cotta and J. Muruzábal. On the learning of Bayesian network graph structures via evolutionary programming. In *Proceedings of the Second European Workshop on Probabilistic Graphical Models*, pages 65–72. Leiden, 2004.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 2012.

H. Cuntz, A. Borst, and I. Segev. Optimization principles of dendritic structure. *Theoretical Biology and Medical Modelling*, 4(1):Article 21, 2008.

H. Cuntz, F. Forstner, A. Borst, and M. Häusser. One rule to grow them all: A general theory of neuronal branching and its practical application. *PLoS Computational Biology*, 6(8):e1000877, 2010.

H. Cuntz, A. Mathy, and M. Häusser. A scaling law derived from optimal dendritic wiring. *Proceedings of the National Academy of Sciences*, 109(27):11014–11018, 2012.

P. Dagum and M. Luby. Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, 60(1):141–153, 1993.

A. Darwiche. A differential approach to inference in Bayesian networks. *Journal of the ACM (JACM)*, 50(3):280–305, 2003.

A. David and L. Pierre. *Hippocampal Neuroanatomy*. Oxford University Press, 2009.

P. Dayan and L. F. Abbott. *Theoretical Neuroscience*. MIT Press, 2001.

G. L. de Haas-Lorentz. *Die Brownsche Bewegung und einige verwandte Erscheinungen*. Springer, 2013.

J. DeFelipe. Neocortical neuronal diversity: Chemical heterogeneity revealed by colocalization studies of classic neurotransmitters, neuropeptides, calcium-binding proteins, and cell surface molecules. *Cerebral Cortex*, 3(4):273–289, 1993.

J. DeFelipe. The dendritic spine story: An intriguing process of discovery. *Frontiers in Neuroanatomy*, 9:Article 14, 2015.

J. DeFelipe and I. Fariñas. The pyramidal neuron of the cerebral cortex: Morphological and chemical characteristics of the synaptic inputs. *Progress in Neurobiology*, 39(6):563–607, 1992.

J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

F. J. Díez. Local conditioning in Bayesian networks. *Artificial Intelligence*, 87(1-2):1–20, 1996.

M. Ding and D. Glanzman. *The Dynamic Brain: An Exploration of Neuronal Variability and its Functional Significance*. Oxford University Press, USA, 2011.

J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In *Proceedings of the Twelfth International Conference in Machine Learning*, volume 12, pages 194–202. Morgan Kaufmann Publishers Inc., 1995.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification. 2nd*. John Willey & Sons, 2001.

G. Elston, R. Benavides-Piccione, A. Elston, J. DeFelipe, and P. Manger. Specialization in pyramidal cell structure in the sensory-motor cortex of the vervet monkey (cercopethicus pygerythrus). *Neuroscience*, 134(3):1057–1068, 2005.

G. N. Elston. Cortex, cognition and the cell: New insights into the pyramidal neuron and prefrontal function. *Cerebral Cortex*, 13(11):1124–1138, 2003.

G. N. Elston, R. Benavides-Piccione, and J. DeFelipe. The pyramidal cell in cognition: a comparative study in human and monkey. *Journal of Neuroscience*, 21(17):RC123, 2001.

R. Etxeberria, P. Larrañaga, and J. M. Picaza. Analysis of the behaviour of genetic algorithms when learning Bayesian network structure from data. *Pattern Recognition Letters*, 18(11): 1269–1273, 1997.

J. Feng. *Computational Neuroscience: A Comprehensive Approach*. CRC Press, 2003.

K. Fernandes and J. S. Cardoso. Discriminative directional classifiers. *Neurocomputing*, 207: 141–149, 2016.

M. Ferrante, M. Migliore, and G. A. Ascoli. Functional impact of dendritic branch-point morphology. *Journal of Neuroscience*, 33(5):2156–2165, 2013.

N. I. Fisher. *Statistical Analysis of Circular Data*. Cambridge University Press, 1995.

N. I. Fisher and A. J. Lee. Regression models for an angular response. *Biometrics*, 48(3): 665–677, 1992.

R. Fisher. Dispersion on a sphere. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 217(1130):295–305, 1953.

M. J. Flores, J. A. Gámez, A. M. Martínez, and J. M. Puerta. GAODE and HAODE: two proposals based on AODE to deal with continuous variables. In *Proceedings of the Twenty-Sixth Annual International Conference on Machine Learning*, pages 313–320. ACM, 2009.

M. J. Flores, J. A. Gámez, A. M. Martínez, and J. M. Puerta. Handling numeric attributes when comparing Bayesian network classifiers: Does the discretization method matter? *Applied Intelligence*, 34(3):372–385, 2011a.

M. J. Flores, A. E. Nicholson, A. Brunskill, K. B. Korb, and S. Mascaro. Incorporating expert knowledge when learning Bayesian network structure: A medical case study. *Artificial Intelligence in Medicine*, 53(3):181–204, 2011b.

M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32(200):675–701, 1937.

N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29:131–163, 1997.

L. D. Fu. A comparison of state-of-the-art algorithms for learning Bayesian network structure from continuous data. Master's thesis, Vanderbilt University, Nashville, Tennesse, 2005.

S. Garcia, J. Luengo, J. A. Sáez, V. Lopez, and F. Herrera. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 25(4):734–750, 2013.

P. H. Garthwaite, J. B. Kadane, and A. O'Hagan. Statistical methods for eliciting probability distributions. *Journal of the American Statistical Association*, 100(470):680–701, 2005.

R. Gatto and S. R. Jammalamadaka. The generalized von Mises distribution. *Statistical Methodology*, 4(3):341–353, 2007.

D. Geiger and D. Heckerman. Learning Gaussian networks. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 235–243. Morgan Kaufmann Publishers Inc., 1994.

I. J. Good. A causal calculus (i). *The British Journal for the Philosophy of Science*, 11(44): 305–318, 1961.

A. L. Gould. A regression technique for angular variates. *Biometrics*, 25(4):683–700, 1969.

I. S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, 7th edition, 2007.

R. Greiner, X. Su, B. Shen, and W. Zhou. Structural extension to logistic regression: Discriminative parameter learning of belief net classifiers. *Machine Learning*, 59(3):297–322, 2005.

D. Grossman and P. Domingos. Learning Bayesian network classifiers by maximizing conditional likelihood. In *Proceedings of the 21st International Conference on Machine Learning*. ACM, 2004.

D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, 1995.

S. Herculano-Houzel. *The Human Advantage: A New Understanding of How Our Brain Became Remarkable*. MIT Press, 2016.

L. D. Hernandez, S. Moral, and A. Salmeron. A Monte Carlo algorithm for probabilistic propagation in belief networks based on importance sampling and stratified simulation techniques. *International Journal of Approximate Reasoning*, 18(1-2):53–91, 1998.

R. Hofmann and V. Tresp. Discovering structure in continuous variables using Bayesian networks. In *Advances in Neural Information Processing Systems*, pages 500–506, 1996.

H. H. Hoos and T. Stützle. *Stochastic Local Search: Foundations and Applications*. Elsevier, 2004.

C.-N. Hsu, H.-J. Huang, and T.-T. Wong. Why discretization works for naive Bayesian classifiers? In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 399–406. Morgan Kaufmann Publishers Inc., 2000.

C.-N. Hsu, H.-J. Huang, and T.-T. Wong. Implications of the Dirichlet assumption for discretization of continuous variables in naive Bayesian classifiers. *Machine Learning*, 53 (3):235–263, 2003.

S. Imoto, T. Goto, and S. Miyano. Estimation of genetic networks and functional structures between genes by using Bayesian networks and nonparametric regression. In *Pacific Symposium on Biocomputing*, volume 7, pages 175–186. World Scientific, 2001.

S. Imoto, S. Kim, T. Goto, S. Aburatani, K. Tashiro, S. Kuhara, and S. Miyano. Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network. *Journal of Bioinformatics and Computational Biology*, 1(02):231–252, 2003.

I. Inza, P. Larrañaga, R. Blanco, and A. J. Cerrolaza. Filter versus wrapper gene selection approaches in DNA microarray domains. *Artificial Intelligence in Medicine*, 31(2):91–103, 2004.

B. Jacobs, M. Schall, M. Prather, E. Kapler, L. Driscoll, S. Baca, J. Jacobs, K. Ford, M. Wainwright, and M. Treml. Regional dendritic and spine variation in human cerebral cortex: A quantitative Golgi study. *Cerebral Cortex*, 11(6):558–571, 2001.

S. R. Jammalamadaka and A. Sengupta. *Topics in Circular Statistics*. World Scientific, 2001.

F. V. Jensen, S. L. Lauritzen, and K. G. Olesen. Bayesian updating in causal probabilistic networks by local computations. *Computational Statistics Quarterly*, 4:269–282, 1990a.

F. V. Jensen, K. G. Olesen, and S. K. Andersen. An algebra of Bayesian belief universes for knowledge-based systems. *Networks*, 20(5):637–659, 1990b.

G. H. John and P. Langley. Estimating continuous distributions in Bayesian classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345. Morgan Kaufmann Publishers Inc., 1995.

N. L. Johnson, S. Kotz, and N. Balakrishnan. *Distributions in Statistics: Continuous Univariate Distributions*, volume 2. Wiley, 1970.

R. A. Johnson and T. E. Wehrly. Some angular-linear distributions and related regression models. *Journal of the American Statistical Association*, 73(363):602–606, 1978.

M. Jones and A. Pewsey. A family of symmetric distributions on the circle. *Journal of the American Statistical Association*, 100(472):1422–1428, 2005.

E. R. Kandel, J. H. Schwartz, T. M. Jessell, S. A. Siegelbaum, and A. J. Hudspeth. *Principles of Neural Science*, volume 4. McGraw-Hill, 2000.

S. Kato. A distribution for a pair of unit vectors generated by Brownian motion. *Bernoulli*, 15(3):898–921, 2009.

S. Kato and M. Jones. A family of distributions on the circle with links to, and applications arising from, Möbius transformation. *Journal of the American Statistical Association*, 105 (489):249–262, 2010.

S. Kato and M. Jones. An extended family of circular distributions related to wrapped Cauchy distributions via Brownian motion. *Bernoulli*, 19(1):154–171, 2013.

S. Kato and M. Jones. A tractable and interpretable four-parameter family of unimodal distributions on the circle. *Biometrika*, 102(1):181, 2015.

S. Kato and A. Pewsey. A Möbius transformation-induced distribution on the torus. *Biometrika*, 102(2):359–370, 2015.

J. T. Kent and D. E. Tyler. Maximum likelihood estimation for the wrapped Cauchy distribution. *Journal of Applied Statistics*, 15(2):247–254, 1988.

J. H. Kim and J. Pearl. A computational model for causal and diagnostic reasoning in inference systems. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, volume 83, pages 190–193. Morgan Kaufmann Publishers Inc., 1983.

Y. Kim, R. Sinclair, N. Chindapol, J. A. Kaandorp, and E. De Schutter. Geometric theory predicts bifurcations in minimal wiring cost trees in biology are flat. *PLoS Computational Biology*, 8(4):e1002474, 2012.

M. J. Kirby and R. Miranda. Circular nodes in neural networks. *Neural Computation*, 8(2): 390–402, 1996.

R. Kohavi. Scaling up the accuracy of naive-Bayes classifiers: A decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, volume 96, pages 202–207. AAAI Press, 1996.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

A. O. Komendantov and G. A. Ascoli. Dendritic excitability and neuronal morphology as determinants of synaptic efficacy. *Journal of Neurophysiology*, 101(4):1847–1866, 2009.

T. J. Koski and J. Noble. A review of Bayesian networks and structure learning. *Mathematica Applicanda*, 40(1):51–103, 2012.

S. Kotz, N. Balakrishnan, and N. L. Johnson. *Continuous Multivariate Distributions, Models and Applications*. John Wiley & Sons, 2004.

J. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1):48–50, 1956.

N. H. Kuiper. Tests concerning random points on a circle. *Indagationes Mathematicae*, 63: 38–47, 1960.

P. Langley and S. Sage. Induction of selective Bayesian classifiers. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 399–406. Morgan Kaufmann Publishers Inc., 1994.

A. U. Larkman. Dendritic morphology of pyramidal neurones of the visual cortex of the rat: I. Branching patterns. *Journal of Comparative Neurology*, 306(2):307–319, 1991.

P. Larrañaga, C. M. Kuijpers, R. H. Murga, and Y. Yurramendi. Learning Bayesian network structures by searching for the best ordering with genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 26(4):487–493, 1996a.

P. Larrañaga, R. Murga, M. Poza, and C. Kuijpers. Structure learning of Bayesian networks by hybrid genetic algorithms. In *Proceedings of the Fifth Conference on Artificial Intelligence and Statistics*, volume 112, pages 165–174. Springer-Verlag, 1996b.

P. Larrañaga, M. Poza, Y. Yurramendi, R. Murga, and C. Kuijpers. Structure learning of Bayesian networks by genetic algorithms: a performance analysis of control parameters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:912 – 926, 1996c.

P. Larrañaga, H. Karshenas, C. Bielza, and R. Santana. A review on evolutionary algorithms in Bayesian network learning and inference tasks. *Information Sciences*, 233:109 – 125, 2013.

S. L. Lauritzen. Propagation of probabilities, means, and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87(420):1098–1108, 1992.

S. L. Lauritzen and F. Jensen. Stable local computation with conditional Gaussian distributions. *Statistics and Computing*, 11(2):191–203, 2001.

J. Lee, W. Chung, and E. Kim. Structure learning of Bayesian networks using dual genetic algorithm. *IEICE Transactions on Information and Systems*, 91(1):32–43, 2008.

I. Leguey, C. Bielza, and P. Larrañaga. Tree-structured Bayesian networks for wrapped Cauchy directional distributions. In *Advances in Artificial Intelligence, Proceedings of the Seventeenth Conference of the Spanish Association for Artificial Intelligence*, volume 9868, pages 207–216. Springer, 2016a.

I. Leguey, C. Bielza, P. Larrañaga, A. Kastanauskaite, C. Rojo, R. Benavides-Piccione, and J. DeFelipe. Dendritic branching angles of pyramidal cells across layers of the juvenile rat somatosensory cortex. *Journal of Comparative Neurology*, 524(13):2567–2576, 2016b.

J. Lemmer and L. Kanal. Propagating uncertainty in Bayesian networks by probabilistic logic sampling. In *Uncertainty in Artificial Intelligence 2*, volume 5, pages 149–163. North-Holland, 1988.

P. Lévy. L'addition des variables aléatoires définies sur une circonférence. *Bulletin de la Société Mathématique de France*, 67:1–41, 1939.

C. Ley and T. Verdebout. *Modern Directional Statistics*. CRC Press, 2017.

S. Lloyd. On a measure of stochastic dependence. *Theory of Probability & its Applications*, 7(3):301–312, 1962.

R. Lockhart and M. Stephens. Tests of fit for the von Mises distribution. *Biometrika*, 72(3): 647–652, 1985.

P. L. López-Cruz, C. Bielza, P. Larrañaga, R. Benavides-Piccione, and J. DeFelipe. Models and simulation of 3D neuronal dendritic trees using Bayesian networks. *Neuroinformatics*, 9(4):347–369, 2011.

P. L. López-Cruz, C. Bielza, and P. Larrañaga. Directional naive Bayes classifiers. *Pattern Analysis and Applications*, 18(2):225–246, 2015.

U. Lund and C. Agostinelli. *Circstats: Circular statistics, from topics in circular statistics*, 2012. URL http://CRAN.R-project.org/package=CircStats.

K. V. Mardia. Statistics of directional data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 37(3):349–393, 1975.

K. V. Mardia and P. E. Jupp. *Directional Statistics*. John Wiley & Sons, 2009.

K. V. Mardia and T. W. Sutton. A model for cylindrical variables with applications. *Journal of the Royal Statistical Society. Series B (Methodological)*, 40(2):229–233, 1978.

K. V. Mardia, G. Hughes, C. C. Taylor, and H. Singh. A multivariate von Mises distribution with applications to bioinformatics. *Canadian Journal of Statistics*, 36(1):99–109, 2008.

H. Markram. The blue brain project. *Nature Reviews Neuroscience*, 7(2):153–160, 2006.

H. Markram. The human brain project. *Scientific American*, 306(6):50–55, 2012.

A. R. Masegosa and S. Moral. An interactive approach for Bayesian network learning using domain/expert knowledge. *International Journal of Approximate Reasoning*, 54(8):1168–1181, 2013.

P. McCullagh. Möbius transformation and Cauchy parameter estimation. *The Annals of Statistics*, 24(2):787–808, 1996.

B. Mihaljević, C. Bielza, R. Benavides-Piccione, J. DeFelipe, and P. Larrañaga. Multi-dimensional classification of gabaergic interneurons with Bayesian network-modeled label uncertainty. *Frontiers in Computational Neuroscience*, 8:Article 150, 2014.

B. Mihaljević, R. Benavides-Piccione, C. Bielza, J. DeFelipe, and P. Larrañaga. Bayesian network classifiers for categorizing cortical GABAergic interneurons. *Neuroinformatics*, 13 (2):193–208, 2015a.

B. Mihaljević, R. Benavides-Piccione, L. Guerra, J. DeFelipe, P. Larrañaga, and C. Bielza. Classifying gabaergic interneurons with semi-supervised projected model-based clustering. *Artificial Intelligence in Medicine*, 65(1):49–59, 2015b.

B. Mihaljevic, C. Bielza, and P. Larrañaga. *bnclassify: Learning discrete Bayesian network classifiers from data*, 2015. URL https://CRAN.R-project.org/package=bnclassify. R package version 0.3.2.

M. Minsky. Steps toward artificial intelligence. *Proceedings of the Institute of Radio Engineers*, 49(1):8–30, 1961.

S. Monti and G. F. Cooper. Learning Bayesian belief networks with neural network estimators. In *Advances in Neural Information Processing Systems*, pages 578–584, 1997.

J. E. Morris and P. Laycock. Discriminant analysis of directional data. *Biometrika*, 61(2): 335–341, 1974.

J. Muruzábal and C. Cotta. A primer on the evolution of equivalence classes of Bayesian-network structures. In *Proceedings of the Eighth International Conference on Parallel Problem Solving from Nature*, pages 612–621. Springer, 2004.

J. W. Myers, K. B. Laskey, and K. A. DeJong. Learning Bayesian networks from incomplete data using evolutionary algorithms. In *Proceedings of the First Annual Conference on Genetic and Evolutionary Computation*, pages 458–465. Morgan Kaufmann Publishers Inc., 1999.

P. Nemenyi. *Distribution-free multiple comparisons*. PhD thesis, Princeton University, 1963.

E. A. Nimchinsky, B. L. Sabatini, and K. Svoboda. Structure and function of dendritic spines. *Annual Review of Physiology*, 64(1):313–353, 2002.

M. Oberlaender, C. P. de Kock, R. M. Bruno, A. Ramirez, H. S. Meyer, V. J. Dercksen, M. Helmstaedter, and B. Sakmann. Cell type–specific three-dimensional structure of thalamocortical circuits in a column of rat vibrissal cortex. *Cerebral Cortex*, 22(10):2375–2391, 2011.

J. D. Park and A. Darwiche. A differential semantics for jointree algorithms. In *Advances in Neural Information Processing Systems*, pages 801–808, 2003.

M. Pazzani and D. Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331, 1997.

M. J. Pazzani. Constructive induction of Cartesian product attributes. In *Feature Extraction, Construction and Selection*, volume 453, pages 341–354. Springer, 1998.

J. Pearl. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29 (3):241–288, 1986.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., 1988.

A. Pérez, P. Larrañaga, and I. Inza. Supervised classification with conditional Gaussian networks: Increasing the structure complexity from naive Bayes. *International Journal of Approximate Reasoning*, 43(1):1–25, 2006.

A. Pérez, P. Larrañaga, and I. Inza. Bayesian classifiers based on kernel density estimation: Flexible classifiers. *International Journal of Approximate Reasoning*, 50(2):341–362, 2009.

F. Pernkopf and J. Bilmes. Discriminative versus generative parameter and structure learning of Bayesian network classifiers. In *Proceedings of the Twenty-Second International Conference on Machine Learning*, pages 657–664. ACM, 2005.

F. Pernkopf and P. OLeary. Floating search algorithm for structure learning of Bayesian network classifiers. *Pattern Recognition Letters*, 24(15):2839–2848, 2003.

Z. Petanjek, M. Judaš, I. Kostović, and H. B. Uylings. Lifespan alterations of basal dendritic trees of pyramidal neurons in the human prefrontal cortex: A layer-specific pattern. *Cerebral Cortex*, 18(4):915–929, 2007.

A. Pewsey. The wrapped stable family of distributions as a flexible model for circular data. *Computational Statistics & Data Analysis*, 52(3):1516–1523, 2008.

A. Pewsey, M. Neuhäuser, and G. D. Ruxton. *Circular statistics in R*. Oxford University Press, 2013.

P.I.N.G. Petilla terminology: Nomenclature of features of gabaergic interneurons of the cerebral cortex. *Nature Reviews. Neuroscience*, 9(7):557–568, 2008.

O. Pourret, P. Naïm, and B. Marcot. *Bayesian networks: A practical guide to applications*. John Wiley & Sons, 2008.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, 2008. URL http://www.R-project.org.

A. Rényi. On the dimension and entropy of probability distributions. *Acta Mathematica Academiae Scientiarum Hungarica*, 10(1-2):193–215, 1959a.

A. Rényi. On measures of dependence. *Acta Mathematica Academiae Scientiarum Hungarica*, 10(3-4):441–451, 1959b.

J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

C. Rojo, I. Leguey, A. Kastanauskaite, C. Bielza, P. Larrañaga, J. DeFelipe, and R. Benavides-Piccione. Laminar differences in dendritic structure of pyramidal neurons in the juvenile rat somatosensory cortex. *Cerebral Cortex*, 26(6):2811–2822, 2016.

M. Romanazzi. Discriminant analysis with high dimensional von Mises-Fisher distributions. In *Eighth Annual International Conference on Statistics*, pages 1–16. Athens Institute for Education and Research, 2014.

T. Romero, P. Larrañaga, and B. Sierra. Learning Bayesian networks in the space of orderings with estimation of distribution algorithms. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(4):607–625, 2004.

Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.

M. Sahami. Learning limited dependence Bayesian classifiers. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, volume 96, pages 335–338. AAAI Press, 1996.

A. V. Samsonovich and G. A. Ascoli. Statistical morphological analysis of hippocampal principal neurons indicates cell-specific repulsion of dendrites from their own cell. *Journal of Neuroscience Research*, 71(2):173–187, 2003.

E. L. Schwartz. *Computational Neuroscience*. MIT Press, 1993.

G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

M. Scutari. Learning Bayesian networks with the bnlearn R package. *Journal of Statistical Software*, 35(3):1–22, 2010.

T. J. Sejnowski, C. Koch, and P. S. Churchland. Computational neuroscience. *Science*, 241 (4871):1299–1306, 1988.

A. SenGupta. On the construction of probability distributions for directional data. *Bulletin of Indian Mathematical Society*, 96(2):139–154, 2004.

A. SenGupta and S. Roy. A simple classification rule for directional data. In *Advances in Ranking and Selection, Multiple Comparisons, and Reliability*, pages 81–90. Springer, 2005.

A. SenGupta and F. I. Ugwuowo. A classification method for directional data with application to the human skull. *Communications in Statistics, Theory and Methods*, 40(3):457–466, 2011.

R. D. Shachter. Evaluating influence diagrams. *Operations Research*, 34(6):871–882, 1986.

R. D. Shachter. Probabilistic inference and influence diagrams. *Operations Research*, 36(4): 589–604, 1988.

R. D. Shachter and C. R. Kenley. Gaussian influence diagrams. *Management Science*, 35(5): 527–550, 1989.

C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

P. Spirtes, C. N. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT Press, 2000.

O. Sporns. *Networks of the Brain*. the MIT Press, 2011.

N. Spruston. Pyramidal neurons: Dendritic structure and synaptic integration. *Nature Reviews Neuroscience*, 9(3):206–221, 2008.

S. Sra. A short note on parameter approximation for von Mises-Fisher distributions and a fast implementation of $i_s(x)$. *Computational Statistics*, 27(1):177–190, 2012.

G. J. Stuart and N. Spruston. Dendritic integration: 60 years of progress. *Nature Neuroscience*, 18(12):1713, 2015.

R. Stufflebeam. Neurons, synapses, action potentials, and neurotransmission, 2008.

J. Su, H. Zhang, C. X. Ling, and S. Matwin. Discriminative parameter learning for Bayesian networks. In *Proceedings of the Twenty-Fifth International Conference on Machine Learning*, pages 1016–1023. ACM, 2008.

H. J. Suermondt and G. F. Cooper. Probabilistic inference in multiply connected belief networks using loop cutsets. *International Journal of Approximate Reasoning*, 4(4):283–306, 1990.

H. J. Suermondt and G. F. Cooper. Initialization for the method of conditioning in Bayesian belief networks. *Artificial Intelligence*, 50(1):83–94, 1991.

Y. Tong. *The Multivariate Normal Distribution*. Springer, 1990.

T. Trappenberg. *Fundamentals of Computational Neuroscience*. Oxford University Press, 2009.

I. Tsamardinos, L. E. Brown, and C. F. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, 2006.

A. Tucker, X. Liu, and A. Ogden-Swift. Evolutionary learning of dynamic probabilistic models with large time lags. *International Journal of Intelligent Systems*, 16(5):621–645, 2001.

J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

G. J. Upton and B. Fingleton. *Categorical and Directional Data*. John Wiley and Sons, 1989.

S. van Dijk and D. Thierens. On the use of a non-redundant encoding for learning Bayesian networks from data with a ga. In *Proceedings of the Eighth International Conference on Parallel Problem Solving from Nature*, pages 141–150. Springer, 2004.

J. van Pelt and H. B. Uylings. The flatness of bifurcations in 3D dendritic trees: An optimal design. *Frontiers in Computational Neuroscience*, 5:Article 54, 2011.

R. von Mises. Über die Ganzzahligkeit der Atomgewichte und verwandte Fragen. *Zeitschrift für Physik*, 19:490–500, 1918.

G. S. Watson. Goodness-of-fit tests on a circle. *Biometrika*, 48(1/2):109–114, 1961.

G. S. Watson and P. Statisticien. *Statistics on Spheres*. Wiley, 1983.

G. S. Watson and E. J. Williams. On the construction of significance tests on the circle and the sphere. *Biometrika*, 43(3/4):344–352, 1956.

T. E. Wehrly and R. A. Johnson. Bivariate models for dependence of angular observations and a related Markov process. *Biometrika*, 67(1):255, 1980.

Q. Wen, A. Stepanyants, G. N. Elston, A. Y. Grosberg, and D. B. Chklovskii. Maximization of the connectivity repertoire as a statistical principle governing the shapes of dendritic arbors. *Proceedings of the National Academy of Sciences*, 106(30):12536–12541, 2009.

N. Wermuth. Linear recursive equations, covariance selection, and path analysis. *Journal of the American Statistical Association*, 75(372):963–972, 1980.

H. Wichterle, D. Gifford, and E. Mazzoni. Mapping Neuronal Diversity One Cell at a Time. *Science*, 341(6147):726–727, 2013.

A. Wintner. On the shape of the angular case of Cauchy's distribution curves. *The Annals of Mathematical Statistics*, 18(4):589–593, 1947.

M. L. Wong, W. Lam, and K. S. Leung. Using evolutionary programming and minimum description length principle for data mining of Bayesian networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(2):174–178, 1999.

C. Wu, E. Ivanova, J. Cui, Q. Lu, and Z.-H. Pan. Action potential generation at an axon initial segment-like process in the axonless retinal AII amacrine cell. *Journal of Neuroscience*, 31 (41):14654–14659, 2011.

Y. Yang and G. I. Webb. Non-disjoint discretization for naive-Bayes classifiers. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 666–673. Morgan Kaufmann Publishers Inc., 2002.

Y. Yang and G. I. Webb. On why discretization works for naive-Bayes classifiers. In *Australasian Joint Conference on Artificial Intelligence*, pages 440–452. Springer, 2003.

E. Yfantis and L. Borgman. An extension of the von Mises distribution. *Communications in Statistics-Theory and Methods*, 11(15):1695–1706, 1982.

R. Yuste. *Dendritic Spines*. The MIT Press, 2010.

Z. Zheng and G. I. Webb. Lazy learning of Bayesian rules. *Machine Learning*, 41(1):53–84, 2000.

D. Zwillinger. *Handbook of Differential Equations*, volume 1. Gulf Professional Publishing, 1998.