DEPARTAMENTO DE INTELIGENCIA ARTIFICIAL

Escuela Técnica Superior de Ingenieros Informáticos
Universidad Politécnica de Madrid

PhD THESIS

THEORETICAL STUDIES ON BAYESIAN NETWORK CLASSIFIERS

Author
Gherardo Varando
MS in Mathematics

PhD supervisors

Concha Bielza
PhD in Computer Science

Pedro Larrañaga
PhD in Computer Science

2018

Thesis Committee

President: Antonio Bahamonde Rionda

External Member: Manuele Leonelli

Member: Francisco Herrera Triguero

Member: José Antonio Gámez Martín

Secretary: Emilio Serrano Fernández

Acknowledgments

First of all I want to express my gratitude to my supervisors, Concha Bielza and Pedro Larrañaga, for constantly supporting and motivating me. Concha and Pedro directed the first step of my young academic career and I will always be thankful to them.

Eva Riccomagno has guided me with incredible availability during and after the three months I had been working with her in the University of Genova. Her advices and suggestions have shaped deeply not only the research in this thesis but also my future career.

My colleagues at the Computational Intelligence Group helped me in numerous ways through the last years. Without them the time spent in the university would have been surely less entertaining.

Finally I want to thank the financial support of the following projects and institutions: Cajal Blue Brain (C080020-09), TIN2013-41592-P and TIN2016-79684-P projects, S2013/ICE-2845-CASI-CAM-CM project, Fundación BBVA grants to Scientific Research Teams in Big Data 2016, European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 604102 (Human Brain Project) and European Union's Horizon 2020 research and innovation programme under grant agreement No. 720270 (HBP SGA1). The Associazione Italiana per l'Intelligenza Artificiale under the Incoming Mobility Grant and the Universidad Politécnica de Madrid under the Programa Propio 2017 for financing the research stay in the University of Genova.

Abstract

Machine learning, as one of the fundamental tools of artificial intelligence, has acquired growing importance in the last decades. The increasing availability of large amounts of data and more computational processing power available at a low price have contributed to the spread of machine learning methods in almost all branches of technology. While a great part of the current research focuses on the creation of new algorithms and methods to tackle different problems, it is widely recognized that formal analysis and theoretical results are necessary to really understand the algorithms employed, their limitations and their capabilities. The work developed in the present thesis is focused on this last aspect of the research in machine learning.

We study Bayesian network classifiers and in general generative classifiers based on probabilistic graphical models. Probabilistic graphical models are widely studied in the statistic literature and in this thesis we analyze them in the context of one of the most basic problem in machine learning, binary classification. Our main result is a description of the implications, for the induced decision functions, of the conditional independence statements holding in the probability model. We will state results both for a wide class of Bayesian network classifiers and for undirected Markov network classifiers.

In particular, we describe the classes of discrimination functions associated with some of the most used Bayesian network classifiers over categorical predictors variables. We obtain polynomials interpolating the induced discrimination functions, and thus representing the corresponding decision functions. Thanks to this characterization we are able to bound the number of decisions representable by Bayesian network classifiers with given structures.

We extend the binary classification results to chain multi-label classifiers, analyzing their expressive power when Bayesian network are used as base models. Finally, we describe an algebraic and geometric approach to study discrimination functions of generative classifiers under general Markov properties. The given approach extends the results for Bayesian network classifiers and introduces an elegant framework, based on finite differences, to study discrimination functions of generative classifiers.

Resumen

En las ultimas décadas, el aprendizaje automático ha adquirido importancia como una de las herramientas fundamentales en inteligencia artificial. El incremento en la disponibilidad de datos y capacidad computacional disponible a bajo coste han contribuido a extender los métodos de aprendizaje automático en casi todas las ramas de la tecnología. Mientras que gran parte de la investigación se centra en el desarrollo de nuevos algoritmos y métodos para tratar diferentes problemas, es ampliamente reconocido que el análisis formal y los resultados teóricos son necesarios para entender los algoritmos empleado, sus limitaciones y sus capacidades. El trabajo desarrollado en esta tesis se centra en éste ultimo aspecto de la investigación en aprendizaje automático.

Estudiamos los clasificadores con redes Bayesianas y en general clasificadores generativos basados en modelos gráficos probabilisticos. Los modelos gráficos probabilisticos han sido y siguen siendo ampliamente estudiados en estadística y en esta tesis los analizamos en el contexto de uno de los problemas más representativos en aprendizaje automático, la clasificación binaria. Nuestro resultado principal es la descripción, tanto para redes Bayesianas como para modelos de Markov no dirigidos, de las implicaciones de las independencias condicionadas en las funciones de decisión asociadas.

En particular, describimos las familias de funciones discriminantes asociadas con las familias de clasificadores con redes Bayesianas más utilizados. Construimos polinomios que interpolan las funciones discriminantes inducidas, describiendo así las funciones de decisión. Gracias a la representación polinomial de las funciones discriminantes somos capaces de acotar el número de decisiones representables por clasificadores con redes Bayesianas.

Extendemos estos resultados a clasificadores en cadena para problemas multi etiqueta, analizando su capacidad expresiva asumiendo que los modelos están basados en redes Bayesianas. Por último, describimos un método algebraico y geométrico para estudiar funciones discriminantes de clasificadores generativos bajo propiedades de Markov generales. El método empleado extiende los resultados obtenido en el caso de las redes Bayesianas y describe un marco formal, basado en diferencias finitas, para estudiar las funciones discriminantes de clasificadores generativos.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The most important and representative problem in machine learning is binary supervised classification, that is, the problem of building a model (classifier) from training data able to successively recognize a simple binary class. Notorious examples of this problem include, but are not limited to spam filtering [Sahami et al., 1998], medical testing for a given disease [Morales et al., 2013] and failure detection in industrial processes [Jung et al., 2018, Varghese et al., 2015].

Bayesian network (BN) classifiers [Bielza and Larrañaga, 2014] are probably the most popular example of generative classifiers and they have been employed successfully in various applications. Bayesian network classifiers present many advantages, as the ability of providing estimation for the posterior class probabilities, the interpretability of the model and the inherent insights into the handled problem that a black-box algorithm is unable to provide. Moreover Bayesian Network can be built easily using a combination of data-driven knowledge and experts' opinions.

Bayesian network classifiers are based on a graphical modeling of the underlying probability distribution [Lauritzen, 1996, Pearl, 1988]. They range from the simplest naive Bayes [Minsky, 1961] classifier, where the predictor variables are assumed to be conditionally independent given the class variable, to the unrestricted Bayesian classifier, where a general form of Bayesian network [Pearl, 1988] is permitted.

While the sound probabilistic setting of graphical models gives solid theoretical foundations to the use of Bayesian network classifiers, these methods, as every generative classifier, carry some degree of confusion on which are the induced *decision functions*. However, the so-called discriminative classifiers, usually entail naturally a description of the decision function employed (e.g., a linear function for logistic regression).

The first rigorous result about the induced decision functions of BN classifiers was given by Minsky [1961], showing that the decision boundary in naive Bayes classifiers with binary predictors is linear (a hyperplane in the Boolean hypercube). Since then some other results were provided but there is not in the literature a general study of discrimination functions induced by generative classifiers under conditional independences. This thesis intends to provide such a framework.

In this thesis we study generative classifiers for binary class and categorical predictor variables. We focus on the expressive power and theoretical properties, firstly, of Bayesian network classifiers and lastly of general generative classifiers under Markov properties. In the next section we state the assumptions, hypotheses and objectives of the present thesis. Then we shortly describe the structure of the manuscript.

1.1 Hypotheses and Objectives

We describe here the assumptions, hypotheses and objectives of the present thesis.

Assumptions

- We will only consider binary classification problems, and as an extension, multi-label problems (seen as multi binary-class problems).
- The predictor variables will be always categorical. Some ideas will be also extended to continuous variables, in this case assumed to be Gaussians.
- We will always assume a probabilistic setting. That is, the class variable and the predictors are considered to be random variables, and we assume that a joint probability distribution exists.

Hypotheses

- It is possible to formally describe the family of decision functions representable by Bayesian network classifiers and in general generative classifiers with Markov assumptions.
- Describing generative classifiers and their properties allow to understand some well-known intuitions in the machine learning community and suggest some ideas for developing new models.

Based on the above hypotheses we formulate the following main objectives of the thesis:

Objectives

- Extend the known results about the expressive power of Bayesian network classifiers to general graphical structures and categorical predictor variables with more than two values.
- Describe a framework to study the expressive power of generative classifiers.
- Apply the results to multi-label methods, such as binary relevance and chain classifiers.
- Formalize the implications of general Markov assumptions on the induced decision functions for generative classifiers.
- Understand the implications on the generative vs. discriminative classifiers and suggest ideas about new methods for generative classifiers.

1.2 Document Organization

The present dissertation is divided into six chapters. The first one is the present introduction.

Background (Chapter 2) contains the main mathematical definitions as well as some well-known results in the literature. It presents a simple introduction to graphs (Sec. 2.1), with useful notations to develop the theory of graphical models (Sec. 2.2). Then, binary classification is presented (Sec. 2.3), with a focus on generative models and in particular Bayesian network classifiers (Sec. 2.3.4).

The following three chapters contain the original research developed during the thesis. For each chapter an individual introduction is given and conclusive sections summarize the results. Chapter 3 contains the main results for decision functions of Bayesian network classifiers. Chapter 4 deals with extensions to multi-label problems, in particular chain classifiers with Bayesian networks. Chapter 5 formulates some results for generative classifiers under the undirected Markov property.

Finally, Conclusions (Chapter 6) summarizes the contributions of this dissertation and suggests some ideas for future research. Moreover the published contributions derived from this work are listed in connection with the corresponding chapters.

Chapter 2

Background

In this chapter we state some basic definitions and some background results about probabilistic graphical models. We also define formally binary classification problems and what we intend with generative classifiers. Moreover we give a brief introduction to Bayesian network classifiers and we review the previous approaches to the study of their expressive power.

2.1 Notations and Basics

We use bold letters, $\mathbf{x}, \mathbf{X}$ or $\mathbf{k}$, to represent elements of a product space, and letters with a subscript to represent the respective components. For example x_2 indicates the second component of $\mathbf{x}$.

We denote random variables with capital letters as $X, Y, Z, X_1, X_2, X_i, X_n$. With bold capital letters we denote vectors of such random variables, and with subscripts we indicate the components of the vector as follow:

$$\mathbf{X} = (X_1, X_2, \ldots, X_n).$$

With x, x_1, x_2 we denote the values of the corresponding random variables X, X_1, X_2. Similarly, $\mathbf{x}$ will denote the value of the random vector $\mathbf{X}$. We denote with $\mathcal{X}_i$ the finite sample space of X_i and with the bold symbol $\boldsymbol{\mathcal{X}}$ the sample space of the random vector $\mathbf{X} = (X_1, \ldots, X_n)$, that is

$$\boldsymbol{\mathcal{X}} = \times_{i=1}^{n} \mathcal{X}_i.$$

If not stated otherwise we will assume the random variable X_i to take value in a categorical sample space, that is a discrete and finite space. Some example of such sample space are the logical values $\{true, false\}$, a set of colors {blue, red , yellow, green}, the bloody type of a person $\{A, B, AB, O\}$, the political party a voter can choose, the type of movie {horror, comedy, drama, ...} or simply the first k integer $[k] = \{1, \ldots, k\}$. In general we will denote the values of X_i as $\mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$. Obviously, we can always consider $\mathcal{X}_i$ embedded in $\mathbb{R}$.

We will use letters P, Q to denote probability distributions of random variables or random vectors. Since we will consider mainly categorical random variables, the probability distributions are obviously specified by their values over the atoms $\{\mathrm{P}(\mathbf{X} = \mathbf{x})\}_{x \in \boldsymbol{\mathcal{X}}}$.

We write $\mathrm{P} > 0$ to denote that the probability P does not assign zero probability to any value, that is,

$$\mathrm{P}(\mathbf{X} = \mathbf{x}) > 0 \quad \text{for all } \mathbf{x} \in \boldsymbol{\mathcal{X}}.$$

Moreover for every $n \in \mathbb{N}$ we denote with $[n]$ the set of the first n positive integers, $[n] = \{1, \dots, n\}$, and for every $A \subseteq [n]$ we denote with $\mathbf{X}_A$ the random vector with components $(X_i)_{i \in A}$. Similarly, $\mathbf{x}_A = (x_i)_{i \in A}$ and $\boldsymbol{\mathcal{X}}_A$ is the sample space of $\mathbf{X}_A$, $\boldsymbol{\mathcal{X}}_A = \times_{i \in A} \mathcal{X}_i$. Analogously, we define the complementary $\mathbf{X}_{-A} = (X_i)_{i \in [n] \setminus A}$, $\mathbf{x}_{-A}$ and $\boldsymbol{\mathcal{X}}_{-A} = \times_{i \notin A} \mathcal{X}_i$.

Given a dataset of observations, $\mathcal{D} = \{\mathbf{x}^j\}_{j \in [N]}$, we define the marginal counts $N_{\mathcal{D}}(\mathbf{X}_A = \mathbf{x}_A)$ as the number of observations out of N such that $\mathbf{x}^j_A = \mathbf{x}_A$.

2.1.1 Conditional Independence

If X, Y, Z are random variables with probability distribution P. We say that X is conditionally independent of Y given Z if for any measurable set A in the sample space of X, there exists one version of the conditional probability $\mathrm{P}(X \in A | Y, Z)$ that is not a function of Y. If X, Y, Z are discrete random variables the conditional independence is equivalent to

$$\mathrm{P}(X = x, Y = y | Z = Z) = \mathrm{P}(X = x | Z = z)\, \mathrm{P}(Y = y | Z = z).$$

When X is conditionally independent of Y given Z we write

$$X \perp\!\!\!\perp Y | Z.$$

Obviously if Z is trivial the conditional independence reduces to the usual independence between random variables.

Similarly we can define conditional independence between random vectors (given random vectors).

2.1.2 Graphs

A graph $\mathcal{G}$ is a pair (V, E), where V is the set of vertices or nodes and E the set of edges. We will always consider *simple* graphs, that is, with no multiple edges and no loops. The set of edges is a subset of the Cartesian product $V \times V$. For $a, b \in V$, if both edges (a, b) and (b, a) are present in E we will call the edge *undirected*, otherwise it will be called *directed*. A graph with just undirected edges is called an *undirected graph* while, obviously a *directed graph* is a graph where all the edges are directed. A directed edge it is also called an *arc*. We will consider just directed or undirected graphs, the context will make it clear the type of graph considered if it is not stated directly.

Given a graph $\mathcal{G} = (V, E)$ and a subset of the vertex set $A \subset V$, the *induced subgraph* $\mathcal{G}_A$ is defined as the graph with vertex set A and edges $\{(a, b)$ s.t. $(a, b) \in E$ and $a, b \in A\}$.

Two nodes in a graph are called *adjacent* or *directly connected* if there is an edge between them. A node a is said to be a parent of a node b in a directed graph if the arc (a, b) is present. Conversely b is said to be a child of a. With $pa(a)$ we denote the subset of *parents* of a.

∂A denotes the *boundary* of a subset of nodes $A \subset V$. It consists of the others nodes of the graph $\mathcal{G}$ that are adjacent to a vertex in A. If $A = a$ we will simply write ∂a.

We say that two nodes a, b are connected if there exists a sequence of nodes $\{a_i\}_{i=0}^k$ such that $a_0 = a$, $a_k = b$ and (a_i, a_{i+1}) or (a_{i+1}, a_i) belongs to E. Such a sequence is called a *path* between a and b.

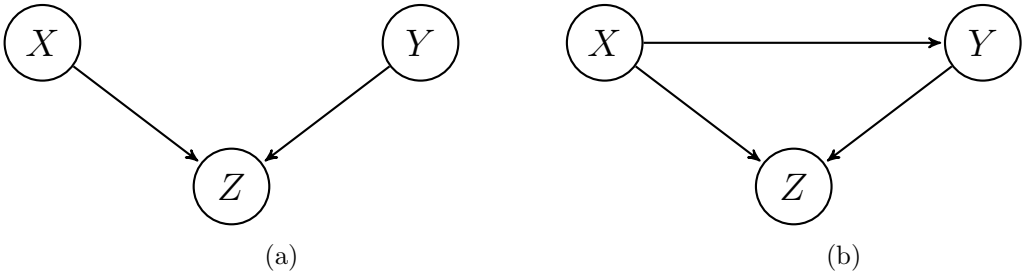


Figure 2.1: Graphical representation of (a) a V-structure and (b) an example which is not a V-structure

Node a is said to lead to b if there exists a *directed path* that goes from a to b, that is, there exists a sequence of nodes $\{a_i\}_{i=0}^{k}$ such that $a_0 = a$, $a_k = b$ and $(a_i, a_{i+1}) \in E$. The *ancestors* of a, denoted $an(a)$, are the nodes that lead to a (but a does not lead to them). The *descendants* of a, $de(a)$, is the set of nodes b such that a leads to b (but b does not lead to a) and the *non-descendants* of a, is the set $nd(a) = V \setminus (de(a) \cup \{a\})$. For $A \subseteq V$ we denote with $an(A) = \cup_{a \in A} an(a) \setminus A$, $de(A) = \cup_{a \in A} de(a) \setminus A$ and $nd(A) = V \setminus (de(A) \cup A)$.

The graph $\mathcal{G}$ is called *complete* if all the nodes are mutually adjacent. If $\mathcal{G}_A$ is complete we say that $A \subset V$ is a complete subset of nodes, or equivalently that A induces a complete subgraph. If a complete subset of nodes $A \subset V$ is maximal with respect to inclusion (there is no $B \supseteq A$ such that B is a complete subset of nodes) A is called a *clique*. With $\mathcal{K}(\mathcal{G})$ we indicate the set of the cliques of $\mathcal{G}$.

A V-structure (or immorality) appears when two parent nodes share the same child, but are not directly connected (Figure 2.1a). If $\mathcal{G}$ is a directed acyclic graph we denote with $\mathcal{G}^m$ the moral graph of $\mathcal{G}$, that is, the undirected graph formed from $\mathcal{G}$ by marrying (connecting) parents and deleting directions.

If the graph $\mathcal{G} = (V, E)$ is undirected and A, B, D are three mutually disjoint subsets of V, we say that D separates A and B in $\mathcal{G}$ if every path from A and B pass through D.

We will always deal with graphs such that the nodes are indexed by a set of random variables, so we will use the same symbols (e.g., X_1, X_2, X_i, C) to denote the random variables and the nodes of the graph. In this case, we will write $\mathbf{X}_{pa(i)}$ for $pa(X_i)$, $\mathbf{X}_{an(i)}$ for $an(X_i)$ and so on.

2.2 Probabilistic Graphical Models

Probabilistic graphical models over discrete random variables are well-studied parametric models. In general they consist of a graph $\mathcal{G}$ where each vertex is associated with a random variable and a joint probability distribution which satisfies some conditional independence statements that can be read from the graph.

Depending on the type of graph we can have different graphical models. The two main and most studied types are Markov network (for undirected graphs) and Bayesian networks (BNs), when the graph is directed and acyclic.

In this section we consider a vector of categorical random variables $\mathbf{X} = (X_1, \ldots, X_n)$ taking values in $\boldsymbol{\mathcal{X}} = \times_i \mathcal{X}_i$ with joint probability distribution P and a graph $\mathcal{G}$ with the

n vertices indexed as the random variables $X_1, \ldots, X_n$. The results we provide in this section are all well-known results in statistics and an extensive treatment can be found in Lauritzen [1996]. In particular we restrict ourselves to discrete sample spaces. In general the results in this section are valid for every sample space given the existence of a density function for the probability P with respect to a product measure over the sample space (e.g. Gaussian distributions).

2.2.1 Markov Models

Assume the graph $\mathcal{G}$ to be undirected. We define the following undirected Markov properties:

(P) **pairwise Markov property**: if for any pair of non-adjacent vertices X_i and X_j we have that

$$X_i \perp\!\!\!\perp X_j | \mathbf{X}_{-\{i,j\}}$$

(L) **local Markov property**: if for any vertex X_i

$$X_i \perp\!\!\!\perp \mathbf{X}_{-D} | \partial X_i \text{ where } D = X_i \cup \partial X_i$$

(G) **global Markov property**: if for any A, B, D disjoint subsets of $[n]$ such that X_D separates X_A and X_B in $\mathcal{G}$,

$$\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B | \mathbf{X}_D$$

Under positivity of the joint probability it is possible to prove that all the three Markov properties are equivalent. Moreover the following well-known result holds [Hammersley and Clifford, 1971, Lauritzen, 1996, Gandolfi and Lenarda, 2017].

Theorem 2.1 (Hammersley-Clifford)**.** *If the joint probability* P *is strictly positive then* P *satisfies the pairwise Markov property with respect to* $\mathcal{G}$ *if and only if it factorizes according to* $\mathcal{G}$*, that is,*

$$\log \mathrm{P}(\mathbf{X} = \mathbf{x}) = \sum_{A \subseteq [n]} \phi_A(\mathbf{x}_A) \quad \textit{s.t. } \phi_A \equiv 0 \textit{ if } \mathcal{G}_A \textit{ is not complete.}$$

The functions ϕ_A are called *interactions*. If the cardinality of A is equal to m we say that ϕ_A is an interaction of order $m-1$.

The proof of Theorem 2.1 (see Lauritzen [1996]) is based on the following combinatorial result.

Lemma 2.1 (Möbius inversion)**.** *Let* ψ *and* ϕ *be functions defined on the set of all subsets of a finite set* Γ *taking values in an Abelian group. Then the following two statements are equivalent:*

(i) For all $A \subseteq \Gamma$, $\psi(A) = \sum_{B \subseteq A} \phi(B)$.

(ii) For all $A \subseteq \Gamma$, $\phi(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} \psi(B)$.

The joint probability P is said to be *Markov with respect to* $\mathcal{G}$ if it is strictly positive and satisfies the pairwise Markov property (P) (or (L) or (G) equivalently). We indicate with $\mathcal{M}(\mathcal{G})$ the class of all probability distributions Markov with respect to $\mathcal{G}$.

The closure of $\mathcal{M}(\mathcal{G})$ under point-wise convergence is the space of *extended Markov probabilities* $\overline{\mathcal{M}}(\mathcal{G})$. We have that also probabilities in $\overline{\mathcal{M}}(\mathcal{G})$ satisfy (P), (L) and (G), as conditional independence is preserved by point-wise limits [Lauritzen, 1996].

It is important to notice that probabilities in $\mathcal{M}(\mathcal{G})$ are identified by the clique marginals. Formally we have the following result:

Lemma 2.2. *Let* $\mathrm{P}, \mathrm{Q} \in \mathcal{M}(\mathcal{G})$ *if for all* $A \in \mathcal{K}(\mathcal{G})$

$$\mathrm{P}(\mathbf{X}_A = \mathbf{x}_A) = \mathrm{Q}(\mathbf{X}_A = \mathbf{x}_A), \quad \forall \mathbf{x}_A \in \boldsymbol{\mathcal{X}}_A,$$

then $\mathrm{P} = \mathrm{Q}$.

Maximum-Likelihood Estimation

Although the maximum-likelihood estimator for extended Markov models exists and is unique it cannot be solved generally in closed form [Lauritzen, 1996] and iterative methods have to be used. The most common method is the *iterative proportional fitting algorithm* (IPF) that consists of iteratively adjusting the marginal of the cliques [Fienberg, 1970]. In particular, for $A \in \mathcal{K}(\mathcal{G})$, we define:

$$T_A \mathrm{P}(\mathbf{X} = \mathbf{x}) = \mathrm{P}(\mathbf{X} = \mathbf{x}) \frac{N_{\mathcal{D}}(\mathbf{X}_A = \mathbf{x}_A)/|\mathcal{D}|}{\mathrm{P}(\mathbf{X}_A = \mathbf{x}_A)}.$$

Let $A_1, \ldots, A_k$ be an ordering of the cliques of $\mathcal{G}$. We define the j-th step of the iterative proportional scaling algorithm as

$$\mathrm{P}^j = T_{A_1} T_{A_2} \cdots T_{A_k} \mathrm{P}^{j-1}.$$

We thus have that for every starting probability $\mathrm{P}^0 \in \mathcal{M}(\mathcal{G})$, P^j converges to the maximum-likelihood estimation in $\overline{\mathcal{M}}(\mathcal{G})$.

2.2.2 Bayesian Networks

We assume now that the graph $\mathcal{G}$ is a directed acyclic graph. In this case we will say that a probability P *recursively factorizes* with respect to $\mathcal{G}$ if

$$\mathrm{P}(\mathbf{X} = \mathbf{x}) = \prod_{i \in [n]} \mathrm{P}(X_i = x_i | \mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)}).$$

If we consider the moral graph $\mathcal{G}^m$ we have:

Lemma 2.3. *If* P *recursively factorizes with respect to* $\mathcal{G}$*, an acyclic directed graph, then it factorizes according to the (undirected) moral graph* $\mathcal{G}^m$*. Thus, it obeys the global, local and pairwise Markov properties relative to* $\mathcal{G}^m$*.*

For directed acyclic graphs we can define the following directed Markov properties:

(DL) **directed local Markov property**: when each variable is conditionally independent of its non-descendants, given its parents:

$$X_i \perp\!\!\!\perp \mathbf{X}_{nd(i)} | \mathbf{X}_{pa(i)}$$

(DG) **directed global Markov property**: if

$$\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B | \mathbf{X}_D,$$

whenever A and B are separated by D in $\left(\mathcal{G}_{an(A\cup B\cup D)}\right)^m$.

We have the following result equivalent to the Hammersley-Clifford theorem (Theorem 2.1),

Theorem 2.2. *Let $\mathcal{G}$ be a directed acyclic graph, and* P *a probability distribution on the sample space $\boldsymbol{\mathcal{X}}$. The following statements are equivalent:*

(i) P *factorizes recursively with respect to $\mathcal{G}$.*

(ii) P *obeys the directed global Markov property for $\mathcal{G}$.*

(iii) P *obeys the directed local Markov property for $\mathcal{G}$.*

Observe that, in contrast to the undirected case, the assumption of positiveness is not needed.

When P satisfies one of the conditions of Theorem 2.2 we say that P is a *directed Markov distribution* with respect to $\mathcal{G}$ or that P satisfies the directed Markov property with respect to $\mathcal{G}$. For every undirected acyclic graph $\mathcal{G}$ and P a directed Markov distribution (with respect to $\mathcal{G}$), the pair $(\mathcal{G}, \mathrm{P})$ is called a *Bayesian network* (BN) and $\mathcal{G}$ is called its structure.

We will denote $\mathcal{BN}(\mathcal{G})$ the set of directed Markov distributions with respect to $\mathcal{G}$.

Maximum-Likelihood Estimation

Estimation of parameters, that is the conditional probability tables of Bayesian networks, is easily done with empirical frequencies. That is,

$$\hat{\mathrm{P}} = \prod_{i\in[n]} \hat{\mathrm{P}}(X_i | \mathbf{X}_{pa(i)}),$$

where

$$\hat{\mathrm{P}}(X_i = x_i | \mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)}) = \frac{N_{\mathcal{D}}(X_i = x_i, \mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)})}{N_{\mathcal{D}}(\mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)})}.$$

Alternatively, to avoid zeros counts, we can use Laplace smoothing of the parameters,

$$\hat{\mathrm{P}}(X_i = x_i | \mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)}) = \frac{N_{\mathcal{D}}(X_i = x_i, \mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)}) + \alpha}{N_{\mathcal{D}}(\mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)}) + \alpha|\mathcal{X}_i|}.$$

2.3 Classification

In this section we introduce binary classification problems, we will define what we intend with classifiers, probabilistic classifiers, generative classifiers and related concepts.

2.3.1 The Binary Classification Problem

The classification problem can be stated simply as the task of *learning*, from a training dataset, a model that is able to classify or discriminate between the two classes.

We will always assume that the class, C, takes values in $\{-1, +1\}$. The choice of the these values is totally arbitrary and this particular choice will become clear in the following.

Following the literature on theory of pattern recognition [Devroye et al., 1996] we define a *classifier* or *decision function* simply as a function $\phi : \boldsymbol{\mathcal{X}} \to \{-1, +1\}$. We will call $\mathcal{C}$ the set of all classifiers over predictor variables taking values in $\boldsymbol{\mathcal{X}}$. Observe now that by our definition a classifier needs always to choose between one of the two classes.

We stress here that with the word *classifier* we indicate the function that is able to classify instances of the predictors and not (as in some literature) the algorithm that produces such a function (that is, the *learning algorithm*).

2.3.2 Probabilistic Classifiers

A probabilistic classifier can be defined as a conditional probability distribution over C given $\mathbf{X}$. That is,

$$\mathrm{P}(C|\mathbf{X} = \mathbf{x}) \in (0, 1) \quad \text{and} \quad \sum_{c} \mathrm{P}(C = c|\mathbf{X} = \mathbf{x}) = 1, \quad \forall \mathbf{x} \in \boldsymbol{\mathcal{X}}.$$

We can define the *induced classifier* with the most probable a posteriori class as follows.

Definition 2.1. *Given* $\mathrm{P}(C|\mathbf{X})$*, a probabilistic classifier over* $\boldsymbol{\mathcal{X}}$*, the induced classifier or the associate decision function is defined as*

$$\phi_{\mathrm{P}}(\mathbf{x}) = \arg \max_{c \in \{-1,+1\}} \mathrm{P}(C = c|\mathbf{X} = \mathbf{x}).$$

That is, the most probable a posteriori class.

Predicting classes with the most probable a posteriori class is usually called the *Bayes classifier* in the literature [Duda et al., 2000, Devroye et al., 1996]. The Bayes classifier is defined as the classifier that attains the minimum error probability, where the error probability of a classifier is defined as the probability of the set of points where the classifiers does not agree with the true class.

Definition 2.2. *Given a classifier* $\phi \in \mathcal{C}$ *and* Q *the joint (usually unknown) probability over* $\mathbf{X}$ *and* C*, the error probability of* ϕ *is*

$$\mathrm{Err}_{\mathrm{Q}}(\phi) = \mathrm{Q}\left(\phi(\mathbf{X}) \neq C\right).$$

Thus, we have that $\mathrm{P}(C|X) = \mathrm{Q}(C|X)$ is the Bayes classifier, that is, $\mathrm{Err}_{\mathrm{Q}}(\phi_{\mathrm{P}})$ is the minimum among the error probabilities of all classifiers.

Since the class variable is binary, and we assumed that $\mathrm{P}(C = c|\mathbf{X} = \mathbf{x}) > 0$, we can completely describe a probabilistic classifier by what we will call the *induced discrimination function*:

Definition 2.3. *Given a probabilistic classifier* $\mathrm{P}(C|\mathbf{X})$*, we define the induced discrimination function as,*

$$f_{\mathrm{P}}(\mathbf{x}) = \log\left(\frac{\mathrm{P}(C=+1|\mathbf{X}=\mathbf{x})}{\mathrm{P}(C=-1|\mathbf{X}=\mathbf{x})}\right) = \log\left(\frac{\mathrm{P}(C=+1|\mathbf{X}=\mathbf{x})}{1-\mathrm{P}(C=+1|\mathbf{X}=\mathbf{x})}\right).$$

It is obvious to observe that the induced classifier is exactly the sign of the induced discrimination function:

$$\phi_{\mathrm{P}} = sign(f_{\mathrm{P}}),$$

where $sign(f)$ is the point-wise sign of the function f.

$$sign(f)(\mathbf{x}) = \begin{cases} +1 & \text{if } f(\mathbf{x}) \geq 0 \\ -1 & \text{if } f(\mathbf{x}) < 0 \end{cases}$$

We can invert Definition 2.3 and obtain that

$$\mathrm{P}(C=+1|\mathbf{X}=\mathbf{x}) = \frac{e^{f_{\mathrm{P}}(\mathbf{x})}}{1+e^{f_{\mathrm{P}}(\mathbf{x})}},$$
$$\mathrm{P}(C=-1|\mathbf{X}=\mathbf{x}) = \frac{1}{1+e^{f_{\mathrm{P}}(\mathbf{x})}}.$$

Thus specifying f_{P} is equivalent to specify the conditional distribution $\mathrm{P}(C|\mathbf{X})$. Observe that the equivalent representation of probabilistic classifiers with discrimination functions is valid only for binary classification problems.

With $\mathcal{F}$ we denote the set of discrimination functions, that is, the set of all real functions over $\boldsymbol{\mathcal{X}}$,

$$\mathcal{F} = \{f : \boldsymbol{\mathcal{X}} \to \mathbb{R}\}.$$

When $\phi = sign(f)$ we say that f sign-represent ϕ and if $\mathcal{F}' \subset \mathcal{F}$ we denote with $sign(\mathcal{F}')$ the image with respect to the $sign$ operator,

$$sign(\mathcal{F}') = \{\phi \in \mathcal{C} \text{ s.t. } \phi = sign(f) \text{ for } f \in \mathcal{F}'\},$$

that is, the set of classifiers that are sign-represented by discrimination functions in $\mathcal{F}'$.

2.3.3 Generative Classifiers

A generative classifier is a model able not only to predict the class values given an instance of predictor variables, but also to generate samples of the predictor variables given a value for the class variable.

Formally, it consists of a joint probability distribution over the predictor and class variables, $\mathrm{P}(\mathbf{X}, C)$. We assume moreover that the probability distribution is strictly positive.

With $\mathcal{P}$ we indicate the set of generative classifiers over $\boldsymbol{\mathcal{X}}$,

$$\mathcal{P} = \{\mathrm{P}(\mathbf{X}, C) > 0 \text{ s.t. } \mathrm{P} \text{ is a probability distribution}\}$$

Since we can compute $\mathrm{P}(C|\mathbf{X})$ from the joint distribution, it is obvious that every generative classifier induces a probabilistic classifier via,

$$\mathrm{P}(C|\mathbf{X}) = \frac{\mathrm{P}(C, \mathbf{X})}{\mathrm{P}(\mathbf{X})}.$$

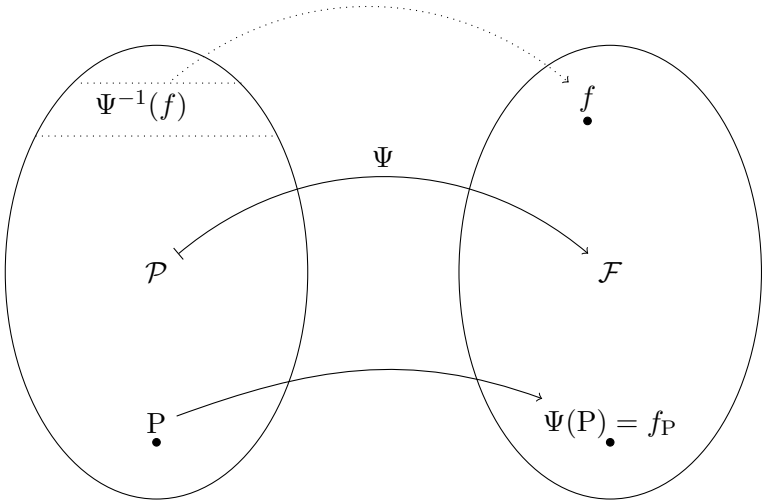


Figure 2.2: The mapping between generative classifiers $\mathcal{P}$ and discrimination function $\mathcal{F}$

Moreover, we can compute the induced discrimination function of P directly from the joint distribution:

$$f_{\mathrm{P}}(\mathbf{x}) = \log\left(\frac{\mathrm{P}(C=+1|\mathbf{X}=\mathbf{x})}{\mathrm{P}(C=-1|\mathbf{X}=\mathbf{x})}\right) = \log\left(\frac{\mathrm{P}(C=+1,\mathbf{X}=\mathbf{x})}{\mathrm{P}(C=-1,\mathbf{X}=\mathbf{x})}\right).$$

Let Ψ be the mapping from $\mathcal{P}$ to $\mathcal{F}$ that assigns the induced discrimination function to every generative classifier P, that is

$$\begin{aligned}\Psi : \mathcal{P} &\to \mathcal{F} \\ \mathrm{P} &\mapsto \Psi(\mathrm{P}) = f_{\mathrm{P}}\end{aligned}$$

For $f \in \mathcal{F}$ the level set (fiber) $\Psi^{-1}(f)$ is the set of generative classifiers that induce f, that is, the set of strictly positive probability distributions P such that $f_{\mathrm{P}} = f$ (Figure 2.2).

Generative vs Discriminative Classifiers

Usually in the literature generative classifiers are seen in opposition to *discriminative* ones. This opposition derives from the learning procedure involved. Generative classifiers are estimated finding the joint probability that is *closest* (usually using the Kullback-Leibler divergence, that is maximizing the likelihood) to the empirical measure of a given sample. Discriminative classifiers algorithms on the contrary try to maximize the conditional-likelihood or equivalently to minimize the error probability (Definition 2.2).

Since the log-likelihood is the sum of the conditional-log-likelihood plus the marginal-log-likelihood it is obvious that maximizing the likelihood is not the best approach to build classifiers if the only goal is to obtain good predictive models for the class variable.

Generative classifiers on the contrary, obtaining a model of the joint probability, are able to deal with missing data easily (through marginalization). Moreover they model the predictors relationships, giving insights in the model behavior.

Ng and Jordan [2001] studied the simplest pair of generative-discriminative equivalent models, namely naive Bayes and logistic regression. They proved that even if logistic regression obtains lower errors asymptotically than naive Bayes, the latter attains faster its limit error and thus they stated that naive Bayes could have an advantage with small sample sizes.

Lasserre et al. [2007] tried to theoretically join the two approaches. They stated that the discriminative approach can be seen as standard maximum-likelihood approach for a different model class, and thus restated the generative-discriminative dichotomy as a different model choice.

2.3.4 Bayesian Network Classifiers

In this section we give a brief account of different models of Bayesian network classifiers. Since we are interested in model descriptions and in the induced decision functions we will not focus on parameter estimation methods and structure search. An extensive survey can be found in Bielza and Larrañaga [2014].

Bayesian network classifiers [Friedman et al., 1997] are probably the most used class of generative classifiers. They consist in modeling, with a Bayesian network, the joint probability distributions P of the predictors and class variables.

Naive Bayes

The naive Bayes [Minsky, 1961] classifier is one of the most used generative classifiers, despite being one of the oldest and surely the simplest model. It relies on the strong independence assumption of the predictor variables being mutually conditionally independent given the class variable C. That is,

$$X_i \perp\!\!\!\perp X_j | C, \quad \forall i \neq j.$$

This fact translates (Theorem 2.2) into the following recursive factorization of probability P,

$$\mathrm{P}(C = c, \mathbf{X} = \mathbf{x}) = \mathrm{P}(C = c) \prod_{i=1}^{n} \mathrm{P}(X_i = x_i | C = c).$$

The graph structure of the naive Bayes is thus given by the every predictor having as parent just the class variable (see Example 2.1).

The factorization permits to estimate probabilities over a large number of predictors with few parameters, making the naive Bayes model competitive with respect to other more complex classifiers especially when the sample size is small.

Example 2.1. *Consider a naive Bayes classifier (structure in Figure 2.3), that is, the simplest BAN, over predictor variables* $X_1 \in \{0, 1, 2\}, X_2 \in \{0, 1\}$. *In this case the joint probability over* (C, X_1, X_2) *is factorized as*

$$\mathrm{P}(C = c, X_1 = x_1, X_2 = x_2) = \mathrm{P}(C = c)\, \mathrm{P}(X_1 = x_1 | C = c)\, \mathrm{P}(X_2 = x_2 | C = c).$$

We consider a uniform prior probability over the class: $\mathrm{P}(C = +1) = 0.5$, $\mathrm{P}(C = -1) = 0.5$, *and conditional probabilities tables given in Table 2.1.*

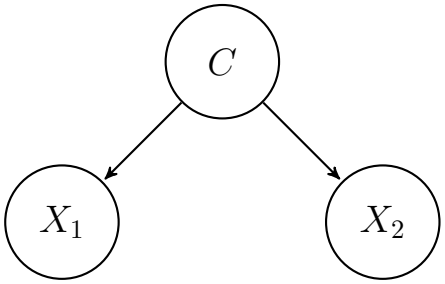


Figure 2.3: Naive Bayes classifier structure in Example 2.1

Table 2.1: Conditional probability tables for X_1 and X_2 in Example 2.1

$P(X_1\|C)$		X_1		
		0	1	2
C	-1	0.3	0.3	0.4
	$+1$	0.1	0.7	0.2

$P(X_2\|C)$		X_2	
		0	1
C	-1	0.5	0.5
	$+1$	0.1	0.9

The induced decision function $\phi_{\mathrm{P}}(x_1, x_2)$, can be computed easily and it is exactly:

$$\phi_{\mathrm{P}}(x_1, x_2) = \begin{cases} -1 & \textit{if } (x_1, x_2) \in \{(0,0),(0,1),(2,0),(2,1)\} \\ +1 & \textit{if } (x_1, x_2) \in \{(1,0),(1,1)\} \end{cases}$$

Augmented Naive Bayes

Starting from the naive Bayes structure, it is then possible to gradually increase the complexity of the graph, and hence the complexity of the obtained factorization.

For every augmented naive Bayes classifier we will call the *predictor subgraph* the subgraph induced by the predictor variables.

Tree augmented naive Bayes Tree Augmented Naive Bayes (TAN) classifier allows a tree structure among the predictors variables [Friedman et al., 1997]. The original algorithm [Chow and Liu, 1968] builds the maximum weighted spanning tree using the conditional mutual information of pairs of predictors as weight.

k-dependence Bayesian classifiers The k-dependence Bayesian model [Sahami, 1996] is an augmented naive Bayes classifier where every predictor is allowed to have a maximum of k parents apart from the class variable.

Bayesian-network augmented naive Bayes The so-called BAN classifier [Friedman et al., 1997, Cheng and Greiner, 1999, 2001] permits a whatsoever Bayesian network as the predictor subgraph. Those are the most general form of Augmented Bayesian network classifiers.

Unrestricted Bayesian Network Classifiers

It is also possible to consider a general BN as a classifier (sometimes called in the literature *unrestricted* BN classifiers), that is without imposing the class variable as parent of all the predictors (see Figure 2.4). But such unrestricted model, although it could be useful for general probability modeling, it is less suitable for classification purposes. Indeed, having predictor variables as parents of the class variable (X_1, X_2 in Figure 2.4) would imply that the contributions of all the parents of the class will not factorize ($\mathrm{P} = \mathrm{P}(X_1)\,\mathrm{P}(X_2)\,\mathrm{P}(C|X_1, X_2)\cdots$ in the example in Figure 2.4). On the other hand, a predictor variable that is not adjacent to the class variable (X_5 or X_6 in the example in Figure 2.4) can be present in two positions:

- As a children of another predictor as X_5 in Figure 2.4. In this case the class variable is independent of X_5 given the value of X_4 (such structures could be useful with missing data).
- As a parent of another predictor as X_6 in Figure 2.4. In this case if X_3 is observed we have that X_6 and C are not independent.

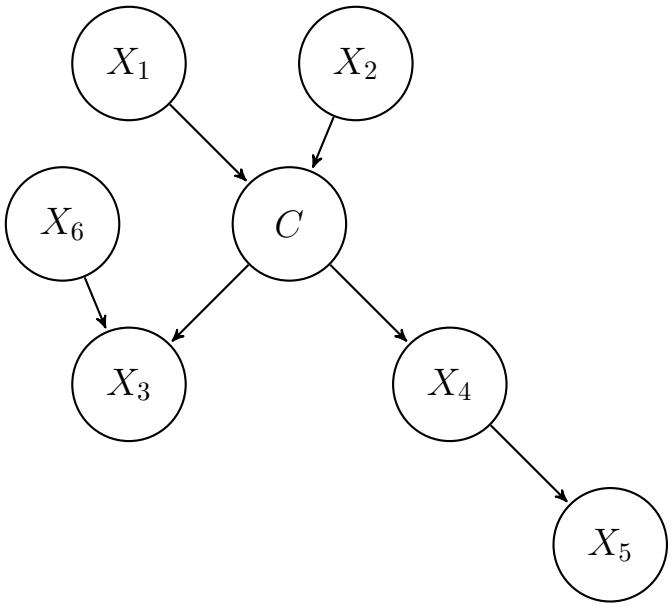


Figure 2.4: Unrestricted BN classifier

The Expressive Power of Bayesian Network Classifiers

The first rigorous result in understanding the limits and capability of Bayesian network classifiers was reported by Minsky [1961], showing that the decision boundary in naive Bayes classifiers with binary predictors is a hyperplane. Since then several other researchers have addressed the problem.

Peot [1996] reviewed Minsky's results about binary predictors and presented some extensions. He mainly discussed the case of naive Bayes with k-valued predictors and predictors dependencies. He also reported an upper bound on the number of linearly

separable dichotomies of the vertices of an n-dimensional cube, consequently bounding the number of decision functions that are representable by naive Bayes classifiers with binary predictors. This is the first example showing that describing the discrimination functions of generative classifiers is useful to obtain informations on their capabilities.

Domingos and Pazzani [1997] studied the optimality of naive Bayes at length and pointed out that, even if the independence assumption among predictors is violated, naive Bayes could achieve optimality under 0-1 loss. Moreover, Domingos and Pazzani [1997] proved a negative result showing that the naive Bayes fails to learn some linearly separable functions when its parameters are estimated with maximum-likelihood, even from a complete and noise-free dataset.

Jaeger [2003] showed that for binary predictors, classifier expressivity is characterized by separability with polynomials of different degrees. Moreover he disagreed with the negative results of Domingos and Pazzani [1997], at least in the interpretation of expressive power, arguing that the inability of the naive Bayes to recognize some linearly separable concepts is a consequence of the training method and not of the model itself, which he showed it is able to generate all possible linear discrimination functions. He than stated that in Domingos and Pazzani [1997] the naive Bayes is not able to learn m-of-n concepts as a consequence of the particular data set used, namely complete and noise-free.

Ling and Zhang [2002] reported negative results for the expressive power of Bayesian networks; they proved that a Bayesian network where each node has at most k parents cannot represent any function containing $(k + 1)$-xors. The results of Ling and Zhang [2002] have the advantages of not being restricted to binary variables and being valid for general BNs. Ling and Zhang [2002] also reported some open question, in particular they conjectured that every function of order m, that is, it does not contains $(m+1)$-xor, can be represented by a BN where every node has at most m parents.

Nakamura et al. [2005] studied the inner product space for Bayesian network classifiers with binary predictors, that is, the smallest Euclidean space that represents the induced class of classifiers. They obtained upper and lower bounds on the dimension of the inner product space and they linked the dimension of the inner product space with the Vapnik-Chervonekis (VC) dimension [Vapnik and Chervonenkis, 1971]. Yang and Wu [2012] studied the case of Bayesian networks with k-valued nodes. They computed the VC dimension for fully connected Bayesian networks and for Bayesian networks without V-structures. In both cases they showed that the VC dimension is equal to the dimension of the inner product space.

Chapter 3

Decision Boundary for Bayesian Network Classifiers

3.1 Introduction

In this chapter we try to generalize the expressivity results [Minsky, 1961, Peot, 1996, Jaeger, 2003] within a unified framework.

In particular we extend and particularized the results of Jaeger [2003]: we show how to build polynomial discrimination functions for any Bayesian augmented naive Bayes classifier with categorical predictors. In absence of V-structures in the predictor subgraph, we prove that the obtained families of polynomials representing the induced decision functions form linear spaces that are representations of the inner product spaces.

We are able to compute the dimensions of those linear spaces and thus of the inner product space extending the results of Nakamura et al. [2005] and Yang and Wu [2012].

Finally, we use the obtained results to bound the number of decision functions representable by BAN classifiers with a given structure.

Chapter Outline

In Section 3.2 we define a polynomial representation of the Iverson bracket [Iverson, 1962] over a finite number of categorical variables and derive the representation of discrete probability functions and of conditional probability tables. We then investigate polynomial representations of discrimination functions induced by Bayesian network classifiers. We look at Bayesian network classifiers in ascending order of complexity: naive Bayes classifiers in Section 3.2.2, tree augmented naive Bayes classifiers in Section 3.2.3, Bayesian network-augmented naive Bayes classifiers in Section 3.2.4 and fully connected Bayesian network classifiers in Section 3.2.5. In Section 3.3 we analyse the expressive power of BAN classifiers. Finally we present the conclusions in Section 3.4.

3.2 Polynomial Threshold Functions for Bayesian Network Classifiers

We develop a method to compute polynomials that represent discrimination functions of Bayesian network classifiers, also called polynomial threshold functions. This method

is an extension of the well-known results on the decision boundary of naive Bayes classifiers [Minsky, 1961, Peot, 1996]. The method is based on the polynomial interpolation of discrete probability functions or equivalently their logarithms. Pistone et al. [2001] gave a more formal and general description of this subject, also addressing applications to Bayesian networks. We will develop this method directly using Lagrange basis polynomials.

3.2.1 Lagrange Interpolation of Discrete Probability

The proofs of the results on the decision boundary in naive Bayes classifiers are based on a representation of the categorical distribution over two values $\{0, 1\}$ in an exponential form, $\mathrm{P}(X = x) = p^x(1-p)^{1-x}$, with $x \in \{0, 1\}$ and $p \in (0, 1)$. We aim to reproduce the same representation for a categorical variable $X \in \mathcal{X} = \{\xi^1, \xi^2, \ldots, \xi^m\} \subset \mathbb{R}$, where the values of variable X are indicated as ξ^j with j as upper index. We consider $\{p(1), \ldots, p(m)\}$ such that $\sum_{j=1}^{m} p(j) = 1$ and, using the Iverson bracket [Iverson, 1962], we write

$$\mathrm{P}(X = x) = \prod_{j=1}^{m} p(j)^{[x=\xi^j]}. \tag{3.1}$$

If $X \in \{0, 1\}$ we could represent $[x = 0]$ as $1 - x$ and $[x = 1]$ as x. If we consider a categorical variable, $X \in \mathcal{X} = \{\xi^1, \xi^2, \ldots, \xi^m\} \subset \mathbb{R}$, we need to find m polynomials $\left\{\ell_j^{\mathcal{X}}\right\}_{j=1}^{m}$ such that

$$\ell_j^{\mathcal{X}}(\xi^j) = 1,$$

and

$$\ell_j^{\mathcal{X}}(\xi^k) = 0 \text{ for every } k \neq j.$$

We easily see that such polynomials exist and have the following form:

$$\ell_j^{\mathcal{X}}(x) = \prod_{k \neq j} \frac{(x - \xi^k)}{(\xi^j - \xi^k)}. \tag{3.2}$$

The polynomials defined in Equation (3.2) are the Lagrange basis polynomials [Abramowitz and Stegun, 1964, Jeffreys and Jeffreys, 1999] over the points in $\mathcal{X}$. These polynomials are m linearly independent polynomials of degree $m-1$, and so they form a basis of polynomials in one variable whose degree is at most $m - 1$. We summarize some properties of these polynomials in the following lemma.

Lemma 3.1. *Let $\mathcal{X}_i = \{\xi_i^1, \xi_i^2, \ldots, \xi_i^{m_i}\} \subset \mathbb{R}$, for $i \in [n]$. For every i define the Lagrange basis, $\left\{\ell_j^{\mathcal{X}_i}(x_i)\right\}$, over $\mathcal{X}_i$ as in Equation (3.2). Then we have*

1. *For every $i \in [n]$, $\left\{\ell_j^{\mathcal{X}_i}(x_i)\right\}_{j=1}^{m_i}$ form a basis of the space of polynomials in x_i of degree $|\mathcal{X}_i| - 1$.*
2. *$\sum_{j_{i_1}=1}^{m_{i_1}} \sum_{j_{i_2}=1}^{m_{i_2}} \cdots \sum_{j_{i_l}=1}^{m_{i_l}} \prod_{s \in I} \ell_{j_s}^{\mathcal{X}_s}(x_s) = \prod_{i \in I} \sum_{j_i=1}^{m_i} \ell_{j_i}^{\mathcal{X}_i}(x_i) = 1$, for every $\mathbf{x} \in \mathbb{R}^I$ and for all $I = \{i_1, \ldots, i_l\} \subseteq [n]$.*
3. *$\prod_{i \in I} \ell_{j_i}^{\mathcal{X}_i}(x_i) = [x_i = \xi_i^{j_i} \ \forall i \in I]$, for every $I \subseteq [n]$, for all $\{j_i\}_{i \in I}$ such that $1 \leq j_i \leq m_i$, and for every $\mathbf{x} \in \times_{i \in I} \mathcal{X}_i$.*

4. $\sum_{j_{i_1}=1}^{m_{i_1}} \sum_{j_{i_2}=1}^{m_{i_2}} \cdots \sum_{j_{i_p}=1}^{m_{i_p}} \prod_{s\in I} \ell_{j_s}^{\mathcal{X}_s}(x_s) = \prod_{i\in I\setminus J} \ell_{j_i}^{\mathcal{X}_i}(x_i)$, *for every* $\mathbf{x} \in \mathbb{R}^I$ *and for all* $J = \{i_i, \ldots, i_p\} \subset I \subseteq [n]$.

Proof. The proof of the above lemma is trivial, and we just outline some points. Point 1 follows from the linear independences of the Lagrange basis polynomials. To prove point 2, we have merely to observe that, since $\left\{\ell_j^{\mathcal{X}_i}\right\}_{j=1}^{m_i}$ is a basis, we have that the polynomial constant 1 admits a unique representation in the considered basis, in particular $1 = \sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i)$. Point 3 follows trivially by substitution. To prove point 4 we apply point 2 as follows,

$$\sum_{j_{i_1}=1}^{m_{i_1}} \sum_{j_{i_2}=1}^{m_{i_2}} \cdots \sum_{j_{i_p}=1}^{m_{i_p}} \prod_{s\in I} \ell_{j_s}^{\mathcal{X}_s}(x_s) = \underbrace{\left(\sum_{j_{i_1}=1}^{m_{i_1}} \sum_{j_{i_2}=1}^{m_{i_2}} \cdots \sum_{j_{i_p}=1}^{m_{i_p}} \prod_{s\in J} \ell_{j_s}^{\mathcal{X}_s}(x_s)\right)}_{=1} \prod_{i\in I\setminus J} \ell_{j_i}^{\mathcal{X}_i}(x_i) = \prod_{i\in I\setminus J} \ell_{j_i}^{\mathcal{X}_i}(x_i).$$

□

If we are given a categorical random variable X over $\mathcal{X} = \{\xi^1, \ldots, \xi^m\}$ whose probability mass function is P, we are able to rewrite Equation (3.1) using the Lagrange basis, as

$$\mathrm{P}(X = x) = \prod_{j=1}^{m} p(j)^{[x=\xi^j]} = \prod_{j=1}^{m} p(j)^{\ell_j^{\mathcal{X}}(x)}, \tag{3.3}$$

where $p(j) = \mathrm{P}(X = \xi^j)$ are the values of the probability mass function over $\mathcal{X}$. Equation (3.3) is a consequence of the identity $[x = \xi^j] = \ell_j^{\mathcal{X}}(x)$ which derives from point 3 of Lemma 3.1 considering $|I| = 1$. More generally, we consider a set of random variables $\{X_1, X_2, \ldots, X_n\}$ such that, for every $i \in [n]$, the variable $X_i \in \mathcal{X}_i = \{\xi_i^1, \xi_i^2, \ldots, \xi_i^{m_i}\}$. If we are given a conditional probability table that represents the probability function $\mathrm{P}(X_1 = x_1 | X_2 = x_2, \ldots, X_n = x_n)$, we can use the Iverson bracket over n variables $x_1, \ldots, x_n$ to describe the conditional distribution of X_1 given $X_2, \ldots, X_n$,

$$\mathrm{P}\left(X_1 = x_1 | X_2 = x_2, \ldots, X_n = x_n\right) = \prod_{(j_1,\ldots,j_n)} p(j_1|j_2, \ldots, j_n)^{[x_i=\xi_i^{j_i}\ \forall i=1,\ldots,n]},$$

where $p(j_1|j_2, \ldots, j_n) = \mathrm{P}(X_1 = \xi_1^{j_1} | X_2 = \xi_2^{j_2}, \ldots, X_n = \xi_n^{j_n})$ are the values of the conditional probability table. Now using point 3 of Lemma 3.1 with $I = [n]$, we get

$$\mathrm{P}\left(X_1 = x_1 | X_2 = x_2, \ldots, X_n = x_n\right) = \prod_{(j_1,\ldots,j_n)} p(j_1|j_2, \ldots, j_n)^{\prod_{i=1}^{m} \ell_{j_i}^{\mathcal{X}_i}(x_i)}. \tag{3.4}$$

3.2.2 Naive Bayes

We consider a naive Bayes classifier (NB) (Figure 3.1) where the predictor variables $X_i \in \mathcal{X}_i$ are conditionally independent given the class variable C. The joint probability distribution factorizes as follows:

$$\mathrm{P}(C = c, X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \mathrm{P}(C = c) \prod_{i=1}^{n} \mathrm{P}(X_i = x_i | C = c). \tag{3.5}$$

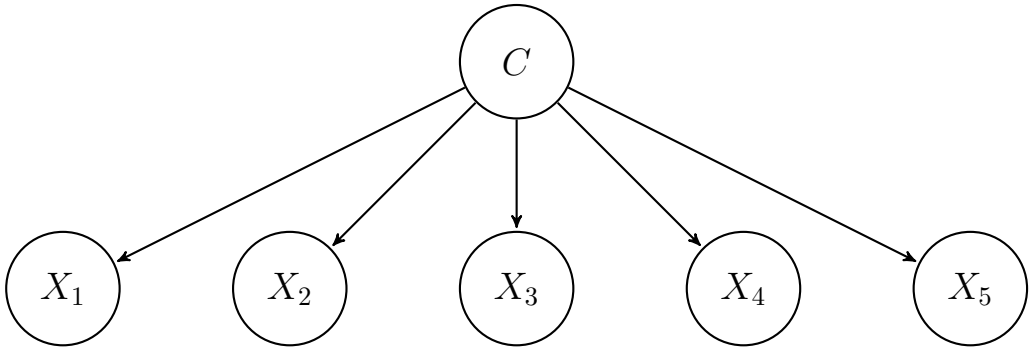
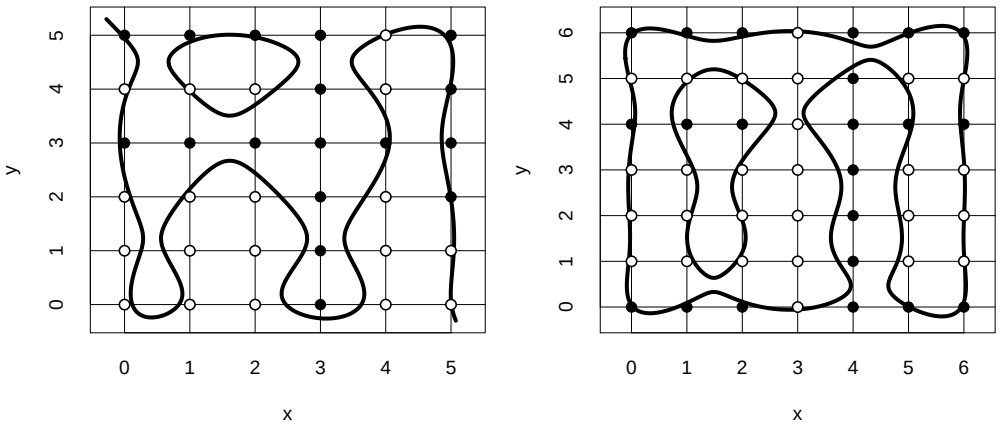


Figure 3.1: Naive Bayes classifier structure with five predictor variables



(a) $X, Y \in \{0, 1, \ldots, 5\}$ (b) $X, Y \in \{0, 1, \ldots, 6\}$

Figure 3.2: Decision boundary for two example, (a) and (b), of naive Bayes classifiers with two categorical variables X, Y. Boundaries are computed as location of zeroes of polynomials built as in Theorem 3.1

If the predictor variables are binary, Minsky [1961] proved that the decision boundaries are hyperplanes. For categorical predictors, the scenario is much more complicated as shown in Figure 3.2.

Theorem 3.1. *A discrimination function $f \in \mathcal{F}$ for a binary classification problem over n categorical variables $X_i \in \mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$, with $|\mathcal{X}_i| = m_i$, is equal over $\boldsymbol{\mathcal{X}}$ to a polynomial of the form $\sum_{i=1}^{n} \left(\sum_{j=1}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(x_i) \right)$ if and only if there exists a naive Bayes classifier, with probability tables without zeros entries, that induces f, where $\ell_j^{\mathcal{X}_i}$ are the Lagrange basis over $\mathcal{X}_i$.*

Proof. We consider a naive Bayes classifier as in Figure 3.1. For every $i \in [n]$ the variable X_i takes values over $\mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$, a subset of $\mathbb{R}$ of cardinality m_i. Thanks to Equation (3.3), we can express, for every value c of the class, the conditional probability

$\mathrm{P}(X_i|C)$ as

$$\mathrm{P}(X_i = x_i|C = c) = \prod_{j=1}^{m_i} p_i(j|c)^{\ell_j^{\mathcal{X}_i}(x_i)},$$

where $p_i(j|c) = \mathrm{P}(X_i = \xi_i^j|C = c)$. If we define $a_i(j|c) = \log(p_i(j|c))$, and assuming that $p_i(j|c) > 0$, we have that

$$\mathrm{P}(X_i = x_i|C = c) = \exp\left(\sum_{j=1}^{m_i} a_i(j|c)\ell_j^{\mathcal{X}_i}(x_i)\right). \tag{3.6}$$

Using this representation we easily find the induced discrimination function for the NB with arbitrary discrete predictor variables. Setting $a = \log(\mathrm{P}(C = +1))$ and $b = \log(\mathrm{P}(C = -1))$, we have that

$$f_{\mathrm{P}}(\mathbf{x}) = \log\left(\mathrm{P}(X_1 = x_1, \ldots, X_n = x_n, C = +1)\right) - \left(\mathrm{P}(X_1 = x_1, \ldots, X_n = x_n, C = -1)\right).$$

Using Equations (3.5) and (3.6) we have that

$$f_{\mathrm{P}}(\mathbf{x}) = \left(a + \sum_{i=1}^{n}\left(\sum_{j=1}^{m_i} a_i(j|+1)\ell_j^{\mathcal{X}_i}(x_i)\right)\right) - \left(b + \sum_{i=1}^{n}\left(\sum_{j=1}^{m_i} a_i(j|-1)\ell_j^{\mathcal{X}_i}(x_i)\right)\right),$$

so the discrimination function for a naive Bayes classifier is

$$f^{NB}(\mathbf{x}) = a - b + \sum_{i=1}^{n}\left(\sum_{j=1}^{m_i} \alpha_i'(j)\ell_j^{\mathcal{X}_i}(x_i)\right), \tag{3.7}$$

where $\alpha'_i(j) = a_i(j|+1) - a_i(j|-1) = \log\left(\frac{\mathrm{P}(X_i=\xi_i^j|C=+1)}{\mathrm{P}(X_i=\xi_i^j|C=-1)}\right)$. We see from Equation (3.7) that the decision function is sign-represented by a polynomial that admits the representation $\sum_{i=1}^{n}\left(\sum_{j=1}^{m_i}\alpha_i(j)\ell_j^{\mathcal{X}_i}(x_i)\right)$. In fact we have that the $a - b = \log\left(\frac{\mathrm{P}(C=+1)}{\mathrm{P}(C=-1)}\right)$ term could be included in the summation using Lemma 3.1, for example with the following choice of coefficient,

$$\alpha_i(j) = \log\left(\frac{\mathrm{P}(X_i = \xi_i^j|C = +1)}{\mathrm{P}(X_i = \xi_i^j|C = -1)}\right) + k_i \log\left(\frac{\mathrm{P}(C = +1)}{\mathrm{P}(C = -1)}\right), \tag{3.8}$$

where $\sum_{i=1}^{n} k_i = 1$. We have proved the *if* part of the theorem.

To prove the *only if* we have just to observe that choosing the conditional probabilities for the predictor variables given the class, $\mathrm{P}(X_i = \xi_i^j|C = c)$, the probability mass for the class $\mathrm{P}(C = +1) = 1 - \mathrm{P}(C = -1)$, and the values of $\{k_i\}_{i=1}^n$ we are able to adjust the coefficients $\alpha_i(j)$ in (3.8) to any possible values in $\mathbb{R}$. For example the

following choices are sufficient

$$\begin{aligned}
&\mathrm{P}(X_i = \xi_i^j | C = -1) = \frac{1}{m_i} \quad \forall i \in [n] \text{ and } j = 1, \ldots, m_i, \\
&\mathrm{P}(X_i = \xi_i^j | C = +1) = \frac{e^{\alpha_i(j)}}{\sum_{j=1}^{m_i} e^{\alpha_i(j)}} \quad \forall i \in [n] \text{ and } j = 1, \ldots, m_i, \\
&k_i = \frac{\log\left(\frac{1}{m_i}\sum_{j=1}^{m_i} e^{\alpha_i(j)}\right)}{\sum_{i=1}^{n} \log\left(\frac{1}{m_i}\sum_{j=1}^{m_i} e^{\alpha_i(j)}\right)} \quad \forall i \in [n], \\
&\log\left(\frac{\mathrm{P}(C=+1)}{\mathrm{P}(C=-1)}\right) = \sum_{i=1}^{n} \log\left(\frac{1}{m_i}\sum_{j=1}^{m_i} e^{\alpha_i(j)}\right).
\end{aligned}$$

□

As a result of Theorem 3.1 we have that a naive Bayes classifier could represent every decision function which is sign-representable by a polynomial of the family

$$\mathcal{F}_{NB} = \left\{ r(\mathbf{x}) = \sum_{i=1}^{n} \left(\sum_{j=1}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(x_i) \right), \alpha_i(j) \in \mathbb{R} \right\}.$$

Only if we fix the prior probability over the class C there are restrictions on the coefficients $\alpha_i(j)$.

Corollary 3.1. *Let $f \in \mathcal{F}$ be a discrimination function for a binary classification problem with n categorical predictor variables $X_i \in \mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\} \subset \mathbb{R}$. The following sentences are equivalent:*

i) f is equal over $\boldsymbol{\mathcal{X}}$ to a polynomial of the form $\sum_{i=1}^{n} \left(\sum_{j=1}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(x_i) \right)$ with $\alpha_i(j)$ such that for every $i = 1, \ldots, n$, there exists $j_{i,1}$ and $j_{i,2}$ such that $\alpha_i(j_{i,1}) < 0$ and $\alpha_i(j_{i,2}) > 0$ or alternatively $e^{\alpha_i(j)} = 1$ for every $j = 1, \ldots, m_i$.

ii) There exists a naive Bayes classifier, with probability tables without zeros entries and with uniform prior probability over the class C, that induces f.

Proof. The corollary follows from (3.8) in proof of Theorem 3.1, it is easy to show that the two conditions are equivalent. □

As we can see, the coefficients $\alpha_i(j)$ are related to the probability model underlying the problem, and are usually estimated from the training set but they do not generally assure the minimization of classification errors. An interesting model to deal with this problem is the weighted naive Bayes classifier [Webb and Pazzani, 1998, Hall, 2007]. Weights are introduced in the probability factorization,

$$\mathrm{P}(C = c | \mathbf{X} = \mathbf{x}) \propto w_c \, \mathrm{P}(C = c) \prod_{i=1}^{n} \left[\mathrm{P}(X_i = x_i | C = c) \right]^{w_i},$$

X_1	$C=-1$	$C=+1$
0	0.3	0.3
1	0.1	0.2
2	0.4	0.1
3	0.1	0.2
4	0.1	0.2

X_2	$C=-1$	$C=+1$
0	0.2	0.4
1	0.1	0.2
2	0.7	0.4

Table 3.1: Conditional probability tables in Example 3.1

$\alpha_1(0) = \log \frac{0.3}{0.3} = 0$	$\alpha_2(0) = \log \frac{0.4}{0.2} = \log 2$
$\alpha_1(1) = \log \frac{0.2}{0.1} = \log 2$	$\alpha_2(1) = \log \frac{0.2}{0.1} = \log 2$
$\alpha_1(2) = \log \frac{0.1}{0.4} = -\log 4$	$\alpha_2(2) = \log \frac{0.4}{0.7} = -\log \frac{7}{4}$
$\alpha_1(3) = \log \frac{0.2}{0.1} = \log 2$	
$\alpha_1(4) = \log \frac{0.2}{0.1} = \log 2$	

Table 3.2: Coefficient computations of the polynomial in Equation (3.9)

and thus the decision function has the same form as in Equation (3.7), but with modified coefficients

$$\alpha_i(j) = w_i \log \frac{\mathrm{P}(X_i = j|C = +1)}{\mathrm{P}(X_i = j|C = -1)}.$$

Note that introducing the weights in the model does not change the form of the polynomial sign-representing the decision functions, so it does not improve the expressive power of the model. Even so, using the weighted model it is possible to search for polynomials that minimize the misclassification and improve accuracy [Zaidi et al., 2013].

Example 3.1. *We consider a naive Bayes classifier with two predictor variables* $X_1 \in \mathcal{X}_1 = \{0, 1, 2, 3, 4\}$ *and* $X_2 \in \mathcal{X}_2 = \{0, 1, 2\}$*. We have a uniform prior probability over the class* C*, that is,* $\mathrm{P}(C = -1) = \mathrm{P}(C = +1) = 0.5$*, and we consider the conditional probability tables for* X_1 *and* X_2 *given in Table 3.1. We can directly build the polynomial threshold functions* $r(x_1, x_2)$ *that sign-represent the decision function induced by this classifier. The related coefficients are* $\alpha_1(j) = \log \frac{\mathrm{P}(X_1=j|C=+1)}{\mathrm{P}(X_1=j|C=-1)}$ *and* $\alpha_2(j) = \log \frac{\mathrm{P}(X_2=j|C=+1)}{\mathrm{P}(X_2=j|C=-1)}$*, and the polynomial* $r(x_1, x_2)$ *is*

$$r(x_1, x_2) = \sum_{j=0}^{4} \alpha_1(j)\ell_j^{\mathcal{X}_1}(x_1) + \sum_{j=0}^{2} \alpha_2(j)\ell_j^{\mathcal{X}_2}(x_2). \tag{3.9}$$

The computations of the coefficients are shown in Table 3.2. We have that the polynomial

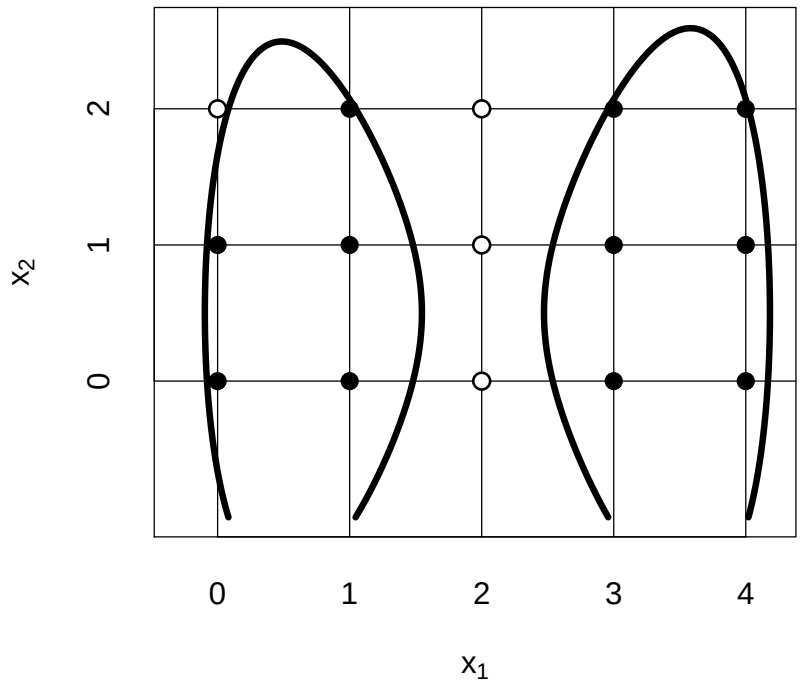


Figure 3.3: Decision boundary for the naive Bayes structure of Example 3.1

threshold function in Equation (3.9), expressed with the Lagrange basis, is

$$\begin{aligned} r(x_1,x_2) =& \frac{x_1(x_1-2)(x_1-3)(x_1-4)}{-6}\log 2 - \frac{x_1(x_1-1)(x_1-3)(x_1-4)}{4}\log 4 \\ &+ \frac{x_1(x_1-1)(x_1-2)(x_1-4)}{-6}\log 2 + \frac{x_1(x_1-1)(x_1-2)(x_1-3)}{24}\log 2 \\ &+ \frac{(x_2-1)(x_2-2)}{2}\log 2 + \frac{x_2(x_2-2)}{-1}\log 2 - \frac{x_2(x_2-1)}{2}\log\frac{7}{4}. \end{aligned}$$

We observe that the above polynomial satisfies the condition of Corollary 3.1, as it should because the prior probability over C is uniform. Figure 3.3 shows the decision boundary induced by $r(x_1, x_2)$.

3.2.3 Tree Augmented Naive Bayes

We now consider a tree augmented naive Bayes (TAN) classifier [Friedman et al., 1997] as shown in Figure 3.4. In this model, a predictor variable $X_i \in \mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$ is allowed to have at most two parents, the class C and an other variable, $X_{pa(i)} \in \mathcal{X}_{pa(i)}$. The joint probability distribution of $(C, X_1, X_2, \ldots, X_n)$ over $\{-1, +1\} \times \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$

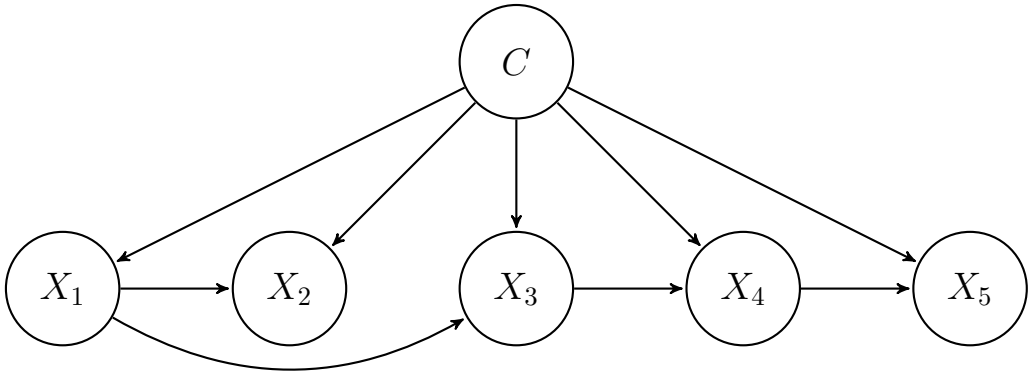


Figure 3.4: Tree augmented naive Bayes classifier structure with five predictor variables

can be factorized according to the Bayesian network theory as

$$\mathrm{P}\left(C=c\right)\prod_{i=1}^{n}\mathrm{P}\left(X_i=x_i|C=c,X_{pa(i)}=x_{pa(i)}\right). \tag{3.10}$$

We can write down a similar representation to the NB case. For each $i=1,\ldots,n$, we apply Equation (3.4) and obtain

$$\mathrm{P}\left(X_i=x_i|C=c,X_{pa(i)}=x_{pa(i)}\right)=\prod_{j=1}^{m_i}\prod_{k=1}^{m_{pa(i)}}p_i(j|c,k)^{\left(\ell_k^{\mathcal{X}_{pa(i)}}(x_{pa(i)})\ell_j^{\mathcal{X}_i}(x_i)\right)}. \tag{3.11}$$

We can now prove, combining Equations (3.10) and (3.11), a result similar to the NB case.

Lemma 3.2. *If f^{TAN} is the discrimination function induced by a TAN for a binary classification problem with n categorical predictor variables $\{X_i\in\mathcal{X}_i\}_{i=1}^n$ and with probability tables without zeros entries, then there exists a polynomial, of the form*

$$\sum_{i=1}^{n}\sum_{j=1}^{m_i}\ell_j^{\mathcal{X}_i}(x_i)\sum_{k=1}^{m_{pa(i)}}\beta_i(j|k)\ell_k^{\mathcal{X}_{pa(i)}}(x_{pa(i)}),$$

that interpolates f^{TAN} over $\boldsymbol{\mathcal{X}}$, where we consider $\sum_{k=1}^{m_{pa(i)}}\beta_i(j|k)\ell_k^{\mathcal{X}_{pa(i)}}(x_{pa(i)})=\beta_i(j)$ when $\mathcal{X}_{pa(i)}=\emptyset$, that is, when class C is the only parent of a node (the root node of the tree).

Proof. The proof is a straightforward computation of the logarithm of Equation (3.10) using Equation (3.11) and the definition $\beta_i(j|k)=\log\left(\frac{p_i(j|+1,k)}{p_i(j|-1,k)}\right)$. The term corresponding to the probability over the class $\log\left(\frac{\mathrm{P}(C=+1)}{\mathrm{P}(C=-1)}\right)$ could be made vanishing into the coefficients of the root node X_t of the tree, using point 2 of Lemma 3.1 with $I=\{t\}$, with the following choice of coefficients

$$\beta_t(j)=\log\left(\frac{p_i(j|+1)}{p_i(j|-1)}\right)+\log\left(\frac{\mathrm{P}(C=+1)}{\mathrm{P}(C=-1)}\right).$$

□

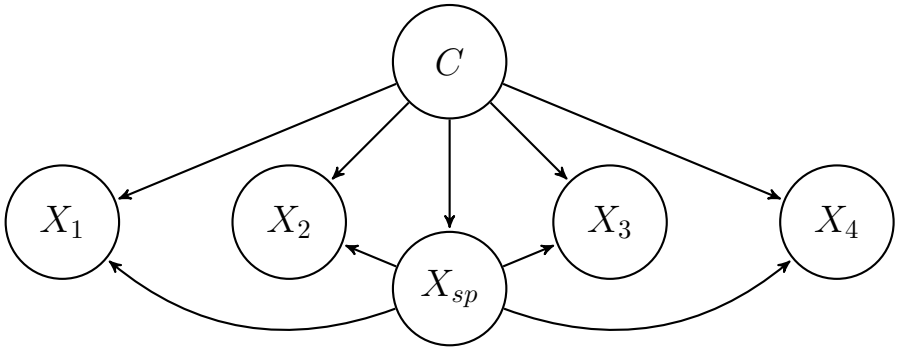


Figure 3.5: SPODE Bayes classifier structure with five predictor variables

A particular case of TAN is the *SuperParent-One-Dependence Estimator* (SPODE) [Keogh and Pazzani, 2002], where all the predictors depend on the same predictor (superparent) (Figure 3.5). The joint distribution factorizes as follows:

$$\mathrm{P}\left(C=c\right)\mathrm{P}\left(X_{sp}=x_{sp}|C=c\right)\prod_{i\neq sp}\mathrm{P}\left(X_i=x_i|C=c,X_{sp}=x_{sp}\right),$$

where X_{sp} stands for the superparent node. In this case, the representation of Lemma 3.2 reduces to

$$f^{SPODE}(\mathbf{x})=sign\left(\sum_{i\neq sp}\sum_{j=1}^{m_i}\ell_j^{\mathcal{X}_i}(x_i)\sum_{k=1}^{m_{sp}}\beta_i(j|k)\ell_k^{\mathcal{X}_{sp}}(x_{sp})\right), \tag{3.12}$$

where f^{SPODE} is the induced discrimination function. If we fix the superparent node, we have a stronger characterization of the induced discrimination functions, the analogue of Theorem 3.1.

Theorem 3.2. *A discrimination function for a binary classification problem over categorical predictor variables is interpolated by a polynomial of the form*

$$\sum_{i\neq sp}\sum_{j=1}^{m_i}\ell_j^{\mathcal{X}_i}(x_i)\sum_{k=1}^{m_{sp}}\beta_i(j|k)\ell_k^{\mathcal{X}_{sp}}(x_{sp}),$$

if and only if it is induced by a SPODE classifier with X_{sp} as the superparent node and with probability tables without zeros entries.

Proof. The *if* part of the theorem is precisely Equation (3.12). To prove the *only if* part we repeat a similar argument as in Theorem 3.1. We observe (Lemma 3.1, point 4, with $J=\{i\}$ and $I=\{i,sp\}$) that for every $i\neq sp$,

$$\ell_k^{\mathcal{X}_{sp}}(x_{sp})=\sum_{j=1}^{m_i}\ell_j^{\mathcal{X}_i}(x_i)\ell_k^{\mathcal{X}_{sp}}(x_{sp}),$$

and so the coefficient $\beta_i(j|k)$ could be seen as

$$\beta_i(j|k)=\log\left(\frac{\mathrm{P}(X_i=j|X_{sp}=k,C=+1)}{\mathrm{P}(X_i=j|X_{sp}=k,C=-1)}\right)+\alpha_i(k),$$

where $\sum_{i \neq sp} \alpha_i(k) = \log\left(\frac{\mathrm{P}(X_{sp}=\xi_{sp}^k|C=+1}{\mathrm{P}(X_{sp}=\xi_{sp}^k|C=-1}\right) + \alpha$ and $\alpha = \log\left(\frac{\mathrm{P}(C=+1)}{\mathrm{P}(C=-1)}\right)$. Then adjusting $\alpha_i(k)$ and α properly we can find a SPODE model, that is, probability distributions over the predictors and the class that induces

$$f = \sum_{i \neq sp} \sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{k=1}^{m_{sp}} \beta_i(j|k) \ell_k^{\mathcal{X}_{sp}}(x_{sp}),$$

for every $\beta_i(j|k) \in \mathbb{R}$. □

Remark 3.1. *We observe that, as for Theorem 3.1, the proof of Theorem 3.2 adds free parameters to the model. For every variable we modify the related coefficients and then we adjust the modifications with the parent coefficients. As in the proof of Theorem 3.1 we are able to use the added parameters to define proper probability distributions, that is to make the defined probability add up to one.*

Remark 3.2. *Results similar to Theorem 3.2 could be proved whenever the structure of the predictor subgraph of a TAN classifier is fixed. We expound no further theorems about TAN classifiers, as, in the next section, we will prove a more general result, of which NB and TAN are special cases.*

Example 3.2. *We look at the SPODE model (see Figure 3.6 for structure) with the superparent node X_{sp}. We consider $X_1 \in \{0,1,2\}$, $X_2 \in \{0,1,2,3\}$ and $X_{sp} \in \{0,1\}$ with conditional probability tables as shown in Table 3.3. The polynomial threshold function $r(x_{sp}, x_1, x_2)$ can be computed directly as specified in Lemma 3.2:*

$$\begin{aligned}
r(x_sp, x_1, x_2) &= (1 - x_{sp}) \log\left(\frac{0.4}{0.8}\right) + x_{sp} \log\left(\frac{0.6}{0.2}\right) \\
&+ (1 - x_{sp}) \left(\frac{(1-x_1)(2-x_1)}{2} \log\left(\frac{0.2}{0.1}\right) + x_1(2-x_1) \log\left(\frac{0.7}{0.1}\right) + \frac{x_1(x_1-1)}{2} \log\left(\frac{0.1}{0.8}\right) \right) \\
&+ x_{sp} \left(\frac{(1-x_1)(2-x_1)}{2} \log\left(\frac{0.7}{0.3}\right) + x_1(2-x_1) \log\left(\frac{0.1}{0.2}\right) + \frac{x_1(x_1-1)}{2} \log\left(\frac{0.2}{0.5}\right) \right) \\
&+ (1 - x_{sp}) \left(\frac{x_2(2-x_2)(3-x_2)}{2} \log\left(\frac{0.3}{0.2}\right) + \frac{x_2(x_2-1)(x_2-2)}{6} \log\left(\frac{0.1}{0.2}\right) \right) \\
&+ x_{sp} \left(\frac{(1-x_2)(2-x_2)(3-x_2)}{6} \log\left(\frac{0.2}{0.5}\right) + \frac{x_2(x_2-1)(3-x_2)}{2} \log\left(\frac{0.5}{0.2}\right) \right).
\end{aligned}$$

We observe that some elements of the Lagrange bases do not appear in $r(x_{sp}, x_1, x_2)$ because the corresponding coefficients are zero, since the conditional probabilities given C are equal.

3.2.4 Bayesian Network-Augmented Naive Bayes

If the predictor subgraph can be a generic Bayesian network, we have a Bayesian network-augmented naive Bayes (BAN) classifier. In this case the joint probability distribution is factorized as follows:

$$\mathrm{P}(C = c) \prod_{i=1}^{n} \mathrm{P}\left(X_i = x_i | C = c, \mathbf{X}_{pa(i)} = \mathbf{x}_{pa(i)}\right), \tag{3.13}$$

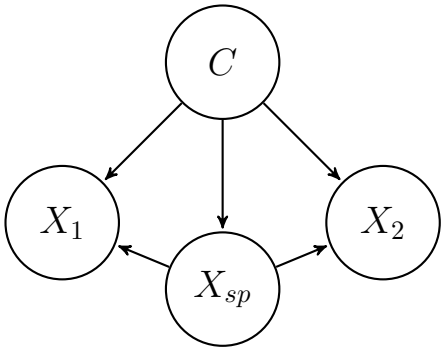


Figure 3.6: SPODE classifier structure, Example 3.2

X_{sp}	$C=-1$	$C=+1$
0	0.8	0.4
1	0.2	0.6

X_1	$C=-1$		$C=+1$	
	$X_{sp}=0$	$X_{sp}=1$	$X_{sp}=0$	$X_{sp}=1$
0	0.1	0.3	0.2	0.7
1	0.1	0.2	0.7	0.1
2	0.8	0.5	0.1	0.2

X_2	$C=-1$		$C=+1$	
	$X_{sp}=0$	$X_{sp}=1$	$X_{sp}=0$	$X_{sp}=1$
0	0.5	0.5	0.5	0.2
1	0.2	0.2	0.3	0.2
2	0.1	0.2	0.1	0.5
3	0.2	0.1	0.1	0.1

Table 3.3: Conditional probability tables in Example 3.2

where $\mathbf{X}_{pa(i)}$ denotes the vector of the parent variables of X_i that are not C. From now on for BAN classifiers we will write $pa(i)$ for the set of indexes defining X_i's parents that are not C and $\mathbb{M}_i = \times_{s \in pa(i)}\{1, \ldots, m_s\}$ for the set of possible configurations of the parents of X_i. Applying the same arguments as in previous sections we can prove the lemma below.

Lemma 3.3. *If f^{BAN} is the discrimination function induced by a BAN classifier for a binary classification problem with n categorical predictors variables $\{X_i \in \mathcal{X}_i \subset \mathbb{R},\ |\mathcal{X}_i| = m_i\}_{i=1}^n$ and with probability tables without zeros entries, then there exists a polynomial of the form*

$$\sum_{i=1}^{n}\sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k}\in\mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s\in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s),$$

which interpolates f^{BAN}, where we write $\sum_{\mathbf{k}\in\mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s\in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s) = \beta_i(j)$ when a variable does not have parents that are not C, that is, $pa(i) = \emptyset$.

Proof. Given a BAN model over predictors $X_i \in \mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$, we define

$$\beta_i(j|\mathbf{k}) = \log\left(\frac{P\left(X_i = \xi_i^j | C = +1, X_s = \xi_s^{k_s},\ \forall s \in pa(i)\right)}{P\left(X_i = \xi_i^j | C = -1, X_s = \xi_s^{k_s},\ \forall s \in pa(i)\right)}\right).$$

Using Equation (3.4) and taking the logarithm of Equation (3.13) we obtain the polynomial representation. The additional constant term due to the prior probability over the class, $\log\left(\frac{\mathrm{P}(C=+1)}{\mathrm{P}(C=-1)}\right)$, could be embedded into the $\beta_i(j|\mathbf{k})$ coefficients using point 2 of Lemma 3.1 as in the proofs of Theorem 3.1 and Lemma 3.2. □

Generally speaking, it is not always possible to prove results similar to Theorem 3.1 or Theorem 3.2 for BAN classifiers, when discrimination functions are completely characterized by sets of polynomials. Like Yang and Wu [2012], we find that problems arise in the presence of V-structures (Figure 2.1a) in the predictor subgraph.

In absence of V-structures we can prove the following result, which extends the previous ones.

Theorem 3.3. *Let $\mathcal{G}$ be a directed acyclic graph with node X_i for $i = 1, \ldots, n$, and let $f \in \mathcal{F}$ be a discrimination function for a binary classification problem over predictor variables $X_i \in \mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$. Suppose that $\mathcal{G}$ does not contain V-structures, then we have that f is interpolated by the following polynomial*

$$r(\mathbf{x}) = \sum_{i=1}^{n}\sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k}\in\mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s\in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s),$$

if and only if f is induced by a BAN classifier whose predictor subgraph is $\mathcal{G}$ and with probability tables without zeros entries.

Proof. We merely have to prove the *only if* because the *if* implication is precisely Lemma 3.3. Given a polynomial of the form

$$r(\mathbf{x}) = \sum_{i=1}^{n}\sum_{j\in\mathcal{X}_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k}\in\mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s\in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s),$$

we have to find a BAN classifier inducing $r(\mathbf{x})$, whose predictor subgraph is $\mathcal{G}$. We just have to define the conditional probability distribution of every variable given its parents, since the structure of the BAN is already fixed by $\mathcal{G}$. For every $i = 1, \ldots, n$, we observe that the subgraph of the parents of X_i is a fully connected Bayesian network, otherwise we will have a V-structure on $\mathcal{G}$. For every i, we can rewrite using point 4 of Lemma 3.1 the i-th addend on the summation,

$$\sum_{j \in \mathcal{X}_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k} \in \mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s) + \sum_{\mathbf{k} \in \mathbb{M}_i} \alpha_i(\mathbf{k}) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s) - \sum_{\mathbf{k} \in \mathbb{M}_i} \alpha_i(\mathbf{k}) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s)$$
$$= \sum_{j \in \mathcal{X}_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k} \in \mathbb{M}_i} (\beta_i(j|\mathbf{k}) + \alpha_i(\mathbf{k})) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s) - \sum_{\mathbf{k} \in \mathbb{M}_i} \alpha_i(\mathbf{k}) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s).$$

Using the *free parameters* $\alpha_i(\mathbf{k})$, it is possible to find for every $\mathbf{k}$, $p_i(j|\mathbf{k}, +1)$ and $p_i(j|\mathbf{k}, -1) \in (0, 1)$ such that

$$\sum_{j=1}^{m_i} p_i(j|\mathbf{k}, +1) = \sum_{j=1}^{m_i} p_i(j|\mathbf{k}, -1) = 1$$
$$\beta_i(j|\mathbf{k}) + \alpha_i(\mathbf{k}) = \log \frac{p_i(j|\mathbf{k}, +1)}{p_i(j|\mathbf{k}, -1)}.$$

To avoid changing the polynomial $r(\mathbf{x})$, we have to subtract

$$\sum_{\mathbf{k} \in \mathbb{M}_i} \alpha_i(\mathbf{k}) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s)$$

from another addend on the summation. Because the parents of X_i are fully connected, we have that among the other addends of $r(\mathbf{x})$, apart from the i-th, there is one product that contains $\prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s)$ and so we just subtract $\alpha_i(\mathbf{k})$ from the related coefficient. Iterating the above procedure for all the nodes of the graph $\mathcal{G}$, we are able to build a probability distribution over $X_1, X_2, \ldots, X_n, C$ that satisfies the Bayesian network structure given by $\mathcal{G}$. More precisely, setting

$$\mathrm{P}\left(X_i = \xi_i^j | C = c, X_s = \xi_s^{k_s}, \forall s \in pa(i)\right) = p_i(j|\mathbf{k}, c),$$

we obtain the target BAN model. □

We observe that the meaning of the representation in Theorem 3.3 is intuitive. If, as usual, we denote by $pa(i)$ the function, dependent on $\mathcal{G}$, that maps each variable X_i to the set of its parents, we have that a new instance $\mathbf{x} = (\xi_1^{j_1}, \ldots, \xi_1^{j_n})$ of the predictors will be classified as $C = +1$ if and only if

$$r(\mathbf{x}) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(\xi_i^{j_i}) \sum_{\mathbf{k} \in \mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s \in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(\xi_s^{j_s})$$
$$= \sum_{i=1}^{n} \ell_{j_i}^{\mathcal{X}_i}(\xi_i^{j_i}) \beta_i(j_i|\{j_s\}_{s \in pa(i)}) \prod_{s \in pa(i)} \ell_{j_s}^{\mathcal{X}_s}(\xi_s^{j_s}) = \sum_{i=1}^{n} \beta_i(j_i|\{j_s\}_{s \in pa(i)}) \geq 0.$$

In other words, every variable X_i, together with its parents $pa(i)$, expresses a degree (positive or negative) $\beta_i(j_i|\{j_s\}_{s \in pa(i)})$ on $\mathbf{x}$, based only on the values of the i-th variable,

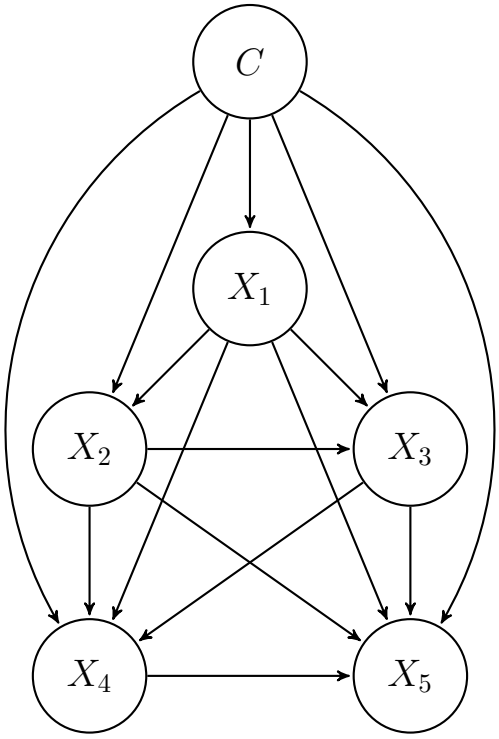


Figure 3.7: FBN classifier structure with five predictor variables

$\xi_i^{k_i}$ and its parent values, $\{\xi_s^{k_s}\ \forall s \in pa(i)\}$. The degrees are summed, and a decision is taken based on the result. The degree expressed by each *coalition* child-parents in the Bayesian network classifier is the logarithm of the ratio between the two probabilities obtained conditioned on the values of the class C,

$$\beta_i(j_i|\{j_s\}_{s\in pa(i)}) = \log \frac{\mathrm{P}(X_i = \xi_i^{j_i}|X_s(i) = \xi_s^{j_s},\, \forall s \in pa(i), C = +1)}{\mathrm{P}(X_i = \xi_i^{j_i}|X_s(i) = \xi_s^{j_s},\, \forall s \in pa(i), C = -1)}.$$

3.2.5 Full Bayesian Networks

When the predictor subgraph is a fully connected Bayesian network (Figure 3.7), that is, a directed acyclic graph with the maximum number of arcs, we have a fully connected Bayesian network classifier (FBN). A FBN can represent any joint probability distribution over $(C, X_1, \ldots, X_n)$ and so it is a classifier able to induce any discrimination function over $\boldsymbol{\mathcal{X}} = \times_{i=1}^{n} \mathcal{X}_i$ whatsoever. We have that the product of the Lagrange bases, $\prod_{i=1}^{n} \ell_{k_i}^{\mathcal{X}_i}(x_i)$, interpolates the Iverson bracket over all the predictors, that is,

$$\prod_{i=1}^{n} \ell_{k_i}^{\mathcal{X}_i}(x_i) = [x_i = \xi_i^{k_i},\, \forall i = 1, \ldots, n].$$

And so the following lemma holds.

Lemma 3.4. *If μ is a probabilistic classifier for a binary class problem with n categorical predictor variables $X_1, \ldots, X_n$ such that $X_i \in \mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\} \subset \mathbb{R}$, $|\mathcal{X}_i| = m_i$, then*

the associated discrimination function, f_μ, is interpolated by a polynomial of the form

$$\sum_{\mathbf{k}\in\mathbb{M}} \gamma_{\mathbf{k}} \prod_{i=1}^{n} \ell_{k_i}^{\mathcal{X}_i}(x_i),$$

where $\mathbb{M} = \times_{i=1}^{n}\{1,\ldots,m_i\}$.

We observe that the coefficients $\gamma_{\mathbf{k}}$ in Lemma 3.4 are the values of the polynomial at point $(\xi_1^{k_1},\xi_2^{k_2},\ldots,\xi_n^{k_n})$, and so $f_\mu(\xi_1^{k_1},\xi_2^{k_2},\ldots,\xi_n^{k_n}) = sign(\gamma_{\mathbf{k}})$. Roughly speaking, a new instance $(\xi_1^{k_1},\xi_2^{k_2},\ldots,\xi_n^{k_n})$ will be classified as $C = +1$ if and only if $\gamma_{\mathbf{k}} > 0$. Moreover we have that the following set

$$\left\{ \sum_{\mathbf{k}\in\mathbb{M}} \gamma_{\mathbf{k}} \prod_{i=1}^{n} \ell_{k_i}^{\mathcal{X}_i}(x_i) \text{ s.t. } \gamma_{\mathbf{k}} \in \mathbb{R} \right\}$$

is a the smallest class of polynomials, which could interpolate every discrimination function (thus it represents every probabilistic classifier), and it is a space of dimension $M = |\mathbb{M}| = \prod_{i=1}^{n} m_i$. From now on we will write

$$\delta_{\mathbf{k}}(\mathbf{x}) = \prod_{i=1}^{n} \ell_{k_i}^{\mathcal{X}_i}(x_i), \tag{3.14}$$

for the $\mathbf{k}$-th element of the canonical basis of $\mathcal{F}$. We call $\{\delta_{\mathbf{k}}\}_{\mathbf{k}\in\boldsymbol{\mathcal{X}}}$ the canonical basis because the sign of the coefficients with respect to this basis is the value of the sign-represented decision function. Lemma 3.4 states that $sign(\mathcal{F}) = \{-1,1\}^{\boldsymbol{\mathcal{X}}}$.

3.3 Expressive Power of Bayesian Network Classifiers

So far, we have seen how to build polynomial that interpolate discrimination functions induced by Bayesian network classifiers. We use now the resulting representation to bound the number of decision functions representable by Bayesian network classifiers. As observed, Lemma 3.4 states that $sign(\mathcal{F}) = \{-1,1\}^{\boldsymbol{\mathcal{X}}}$. We now study NB, SPODE and BAN through the families of discrimination functions representable with the associated polynomials. Moreover, we embed those families in $\mathcal{F}$ with the canonical bases. For predictor variables $X_i \in \mathcal{X}_i = \{\xi_i^1,\ldots,\xi_i^{m_i}\}$, $i = 1,\ldots,n$, for every $sp \in \{1,\ldots,n\}$ and a directed acyclic graph $\mathcal{G}$ without V-structures we define

$$\mathcal{F}_{NB} = \left\{ r(\mathbf{x}) = \sum_{i=1}^{n} \left(\sum_{j=1}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(x_i) \right) \text{ s.t. } \alpha_i(j) \in \mathbb{R} \right\}, \tag{3.15}$$

$$\mathcal{F}_{sp} = \left\{ r(\mathbf{x}) = \sum_{i\neq sp} \sum_{j=1}^{m_i} \sum_{k=1}^{m_{sp}} \beta_i(j|k) \ell_k^{\mathcal{X}_{sp}}(x_{sp}) \ell_j^{\mathcal{X}_i}(x_i) \text{ s.t. } \beta_i(j|k) \in \mathbb{R} \right\}, \tag{3.16}$$

$$\mathcal{F}_{\mathcal{G}} = \left\{ r(\mathbf{x}) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k}\in\mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s\in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s) \text{ s.t. } \beta_i(j|\mathbf{k}) \in \mathbb{R} \right\}, \tag{3.17}$$

where $pa(i)$ is a function that maps every i into the set of parents of X_i in the directed acyclic graph $\mathcal{G}$, and $\mathbb{M}_i = \times_{s\in pa(i)}\{1,\ldots,m_s\}$. The families $\mathcal{F}_{NB}$, $\mathcal{F}_{sp}$ and $\mathcal{F}_{\mathcal{G}}$ are the

sets of polynomials sign-representing the decision functions induced by naive Bayes classifier, SPODE classifier and BAN classifier, respectively. Hence $sign(\mathcal{F}_{NB})$, $sign(\mathcal{F}_{sp})$ and $sign(\mathcal{F}_{\mathcal{G}})$ are the sets of decision functions induced by naive Bayes, SPODE and BAN classifiers, respectively. Obviously, we have that

$$\mathcal{F}_{NB} \subset \mathcal{F}_{\mathcal{G}} \subset \mathcal{F},$$

and

$$sign(\mathcal{F}_{NB}) \subset sign(\mathcal{F}_{\mathcal{G}}) \subset sign(\mathcal{F}) = \{-1, +1\}^{\boldsymbol{\mathcal{X}}}.$$

We can prove that the above sets are indeed subspaces of $\mathcal{F}$ and we can compute their dimensions.

Lemma 3.5. *$\mathcal{F}_{NB}$ is a subspace of $\mathcal{F}_{FBN}$ of dimension $\sum_{i=1}^{n} m_i - n + 1$.*

Proof. Obviously $\mathcal{F}_{NB} = \left\{ p(\mathbf{x}) = \sum_{i=1}^{n} \left(\sum_{j=1}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(x_i) \right), \alpha_i(j) \in \mathbb{R} \right\}$ is a subspace of $\mathcal{F}$. The union of the Lagrange bases over different variables is not a basis, because for each $i = 1, \ldots, n$ we have that

$$1 = \sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \text{ for every } x_i \in \mathbb{R}.$$

So for every i, we can define

$$\mathcal{B}_i = \left\{ \bigcup_{j=2}^{m_i} \{ l_j^{\mathcal{X}_i}(x_i) \} \right\} \cup \{e_0\},$$

where e_0 is the polynomial constant 1, and we find that $\mathcal{B}_i$ is a basis of polynomials in x_i of degree $|\mathcal{X}_i| - 1 = m_i - 1$, equivalent to the Lagrange basis over $\mathcal{X}_i$. Then, we have that

$$\mathcal{B} = \bigcup_{i=1}^{n} \mathcal{B}_i = \bigcup_{i=1}^{n} \bigcup_{j=2}^{m_i} \left\{ l_j^{\mathcal{X}_i}(x_i) \right\} \cup \{e_0\}$$

generates the subspace $\mathcal{F}_{NB}$. We prove that $\mathcal{B}$ is in fact a basis of $\mathcal{F}_{NB}$. We have to prove that the elements of $\mathcal{B}$ are linearly independent. We consider

$$p(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=2}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(x_i) + \alpha_0 e_0 = 0, \, \forall (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n.$$

If, as usual, $\mathcal{X}_i = \{\xi_i^1, \ldots, \xi_i^{m_i}\}$, let us consider $p(x_1, \ldots, x_n)$ evaluated in $(\xi_1^1, \xi_2^1, \ldots, \xi_n^1)$,

$$0 = p(\xi_1^1, \xi_2^1, \ldots, \xi_n^1) = \sum_{i=1}^{n} \sum_{j=2}^{m_i} \alpha_i(j) \ell_j^{\mathcal{X}_i}(\xi_i^1) + \alpha_0 e_0 = \alpha_0,$$

since $\ell_j^{\mathcal{X}_i}(\xi_i^1) = 0$ for every $j \neq 1$. And so $\alpha_0 = 0$. We now evaluate $p(\cdot)$ over $(\xi_1^j, \xi_2^1, \ldots, \xi_n^1)$ and we have that, for every $j = 2, \ldots, m_i$,

$$0 = p(\xi_1^j, \xi_2^1, \ldots, \xi_n^1) = \alpha_1(j),$$

since $\ell_j^{\mathcal{X}_1}(\xi_1^j) = 1$ for every $j = 2, \ldots, m_1$. We repeat the above argument for every variable x_i, $i = 1, \ldots, n$ and we obtain $\alpha_i(j) = 0$ for every $i = 1, \ldots, n$ and every $j = 2, \ldots, m_i$. We have proved that the elements of $\mathcal{B}$ generate $\mathcal{F}_{NB}$ and are linearly independent, so they form a basis of $\mathcal{F}_{NB}$. Consequently we obtain

$$\dim(\mathcal{F}_{NB}) = |\mathcal{B}| = \sum_{i=1}^{n} m_i - n + 1.$$

□

Analogously we can prove, in the general case, the following lemma,

Lemma 3.6. *For every Bayesian network classifier without V-structures in the predictor subgraph $\mathcal{G}$, the set $\mathcal{F}_{\mathcal{G}}$ is a subspace of $\mathcal{F}$ of dimension*

$$\sum_{i=1}^{n} \left((m_i - 1) \prod_{s \in pa(i)} m_s \right) + 1.$$

And, in the particular case of $SPODE$, we have,

Lemma 3.7. *For every $sp = 1, \ldots, n$, the set $\mathcal{F}_{sp}$ is a subspace of $\mathcal{F}$ of dimension*

$$m_{sp} \left(1 - n + \sum_{i \neq sp} m_i \right).$$

We now consider the space $\mathcal{F}$ with respect to the canonical basis given by Equation (3.14). With respect to this coordinate system we have that each orthant represents a decision function. We know that the number of orthants of an M-dimensional space is 2^M, the number of decision functions over a set of cardinality M. Since we now have a bijection between orthants in $\mathcal{F}$ and decision functions over $\boldsymbol{\mathcal{X}}$, in order to compute how many decision functions are representable by a class of Bayesian network classifier (NB, SPODE or BAN) we merely have to count the number of orthants in $\mathcal{F}$ intersected by the corresponding subspaces ($\mathcal{F}_{NB}$, $\mathcal{F}_{sp}$, $\mathcal{F}_{\mathcal{G}}$).

Theorem 3.4 (Flatto, 1970)**.** *A d-dimensional subspace in an M-dimensional space intersects at most $C(M,d) = 2\sum_{k=0}^{d-1} \binom{M-1}{k}$ orthants with equality if and only if it is in general position.*

Definition 3.1. *A d-dimensional subspace V of $\mathbb{R}^M$ is in general position if the M subspaces $V \cap H_i$, where $H_i = \{\mathbf{x} \in \mathbb{R}^n \text{ s.t. } x_i = 0\}$ are hyperplanes of V in general position, that is, all the intersections of d of such hyperplanes are the zero vector. Precisely, for all $J \subset \{1, \ldots, M\}$ such that $|J| = d$ we have that $\bigcap_{j \in J} (V \cap H_j) = \mathbf{0}$.*

Applying Theorem 3.4 to our case, we find that the space $\mathcal{P}^{FBN}$ is minimal in the following sense.

Corollary 3.2. *If V is a d-dimensional subspace of $\mathcal{F}$, then $|sign(V)| \leq C(M,d)$, where $M = dim(\mathcal{F})$ and equality holds if and only if V is in general position with respect to the canonical basis of $\mathcal{F}$.*

As a first result of Corollary 3.2 we have that the space $\mathcal{F}$ is the *smallest* vectorial space of polynomials in $x_1, \ldots, x_n$ that sign-represents every decision function over $\boldsymbol{\mathcal{X}}$, that is, there is not a space V of polynomials in $x_1, \ldots, x_n$ with degrees in each variable x_i that are less or equal than $m_i - 1$ such that $sign(V) = \{-1, +1\}^{\boldsymbol{\mathcal{X}}}$ and $dim(V) < dim(\mathcal{F})$. This justifies the choice of $\mathcal{F}$ as the space to study the polynomial families defined in Equations (3.15), (3.16) and (3.17). Next, we can use Corollary 3.2 combined with Lemma 3.6 to upper bound the number of decision functions that are sign-representable by BAN classifiers with a fixed predictor subgraph $\mathcal{G}$ not containing V-structures.

Corollary 3.3. *Consider a BAN classifier over predictor variables $X_i \in \mathcal{X}_i$, $|\mathcal{X}_i| = m_i$ for every $i = 1, \ldots, n$. Moreover suppose that the predictor subgraph $\mathcal{G}$ does not contain V-structures. Then we have*

$$2^d \leq |sign(\mathcal{F}_{\mathcal{G}})| \leq C(M, d) = 2 \sum_{k=0}^{d-1} \binom{M-1}{k},$$

where $d = \sum_{i=1}^{n} \left((m_i - 1) \prod_{s \in pa(i)} m_s \right) + 1$ and $M = \prod_{i=1}^{n} m_i$.

Peot [1996] observed that naive Bayes could only represent a fraction of dichotomies (binary decision) on binary predictors, and that this fraction goes to zero as the number of predictors increase, we extend this observation to BAN classifier without V-structures as follows.

Corollary 3.4. *We consider, for every $n \in \mathbb{N}$, classification problems with predictors $X_i \in \mathcal{X}_i \subset \mathbb{R}$, $|\mathcal{X}_i| = m_i$ for $i = 1, \ldots, n$. For every n, let $\mathcal{G}_n$ be a directed acyclic graph over the predictor variables, not containing V-structures. Suppose moreover that if $pa_n(i)$ are the functions that map every X_i into the set of parents in the graph $\mathcal{G}_n$,*

$$|pa_n(i)| \leq K \quad \forall n \in \mathbb{N} \text{ and } i \in \{1, \ldots, n\},$$

then we have that

$$\lim_{n \to \infty} \frac{\left| sign\left(P_{\mathcal{G}_n}^{BAN}\right) \right|}{\left| \{-1, +1\}^{\boldsymbol{\mathcal{X}}(n)} \right|} = \lim_{n \to \infty} \frac{\left| sign\left(P_{\mathcal{G}_n}^{BAN}\right) \right|}{2^{|\boldsymbol{\mathcal{X}}(n)|}} = 0,$$

where $\boldsymbol{\mathcal{X}}(n) = \times_{i=1}^{n} \mathcal{X}_i$. In other words, the fraction of decision functions representable by BAN classifiers, with a fixed maximum number of parents for each variable, becomes vanishingly small by increasing the number of predictors.

Proof. For every $n \in \mathbb{N}$, we apply Corollary 3.3 and we obtain

$$|sign\left(\mathcal{F}_{\mathcal{G}_n}\right)| \leq C\left(M(n), d(n)\right) = 2 \sum_{k=0}^{d(n)-1} \binom{M(n)-1}{k},$$

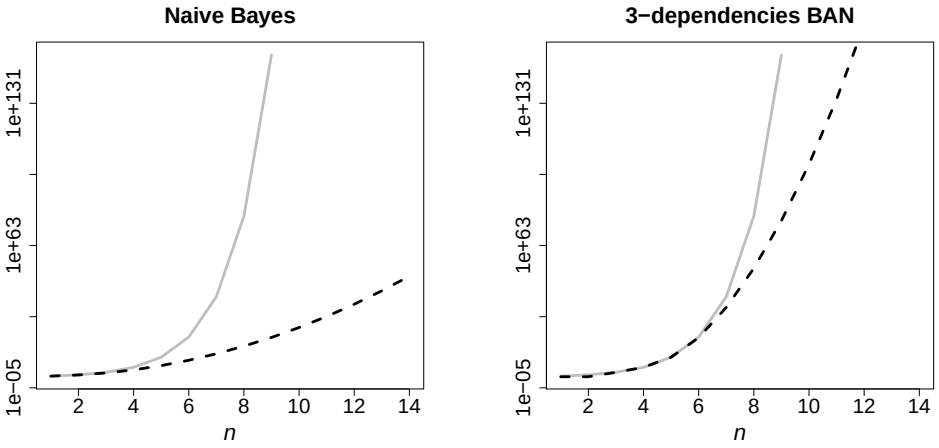
where $d(n) = \sum_{i=1}^{n} \left((m_i - 1) \prod_{s \in pa(i)} m_s \right) + 1$ and $M(n) = |\boldsymbol{\mathcal{X}}(n)| = \prod_{i=1}^{n} m_i$. We observe now that, as $n \to \infty$,

$$\frac{d(n)}{M(n)} \to 0$$

and thus,

$$\frac{C(M(n), d(n))}{2^{M(n)}} \to 0,$$

which proves the statement. □



(a) (b)

Figure 3.8: Total number of decision functions over n binary predictors (solid gray) and the bounding $C(M, d)$ of Corollary 3.3 (dashed black) for NB classifiers (a) and for 3-dependence BAN classifiers (b)

3.4 Conclusions

In this chapter we have shown how to build polynomial threshold functions related to Bayesian network classifiers. Our results reveal connections between the algebraic structure of the decision functions induced by BN classifiers and the topology of the structure of the predictor subgraph. In absence of V-structures in the predictor subgraph we have also proved that the specific polynomial representation fully characterized the type of Bayesian network classifier. By representing classifiers by polynomial threshold functions, we can obtain bounds on the number of decision functions which can be induced by Bayesian network classifiers with a given structure. The bounding does not hold in presence of V-structures in the predictor subgraph. Strong characterizations of induced decision functions cannot be proven due to the conditional independence of V-structure. The polynomial representation of the discrimination functions of BAN classifiers implies the results of Roos et al. [2005] that connect Bayesian network classifiers and generalized logistic regression models. Moreover we observe that the obtained polynomial representation permits to easily prove the results of Ling and Zhang [2002] for BAN classifiers without V-structures.

The bounds points to the fact, already conjectured by Peot [1996] for naive Bayes, that if we fix the maximum number of parents in a Bayesian network classifier, the type of classifier considered is not *scalable*, in other words, more complex classifiers are expected to perform better when dealing with a large number of predictor variables.

Moreover, the resulting bounds for the number of decision functions representable are strictly upper bounds since the subspaces generated by the different Bayesian networks considered are not in general position. What happens in the case of subspaces not in general position? Clearly we have to define some other property to characterize the *position* of a subspace with respect to orthants in some given basis and try to count the number of such intersected orthants. With similar geometric results we will be able to precisely count the number of decision functions representable by a given Bayesian

network classifier, and we will be able to compute the gain in expressibility from simple to more complicated Bayesian network classifiers.

Chapter 4

Decision Functions for Chain Classifiers Based on Bayesian Networks for Multi-Label Classification

4.1 Introduction

We consider a multi-label classification problem [Zhang and Zhou, 2014, Tsoumakas and Katakis, 2007] over categorical predictors, that is, mapping every instance $\mathbf{x} = (x_1, \ldots, x_n)$ to a subset of h labels:

$$\boldsymbol{\mathcal{X}} = \mathcal{X}_1 \times \cdots \times \boldsymbol{\mathcal{X}}_n \to Y \subseteq \mathcal{Y} = \{y_1, \ldots, y_h\},$$

where $\mathcal{X}_i \subset \mathbb{R}$, $|\mathcal{X}_i| = m_i < \infty$. As usually the problem could be transformed into a multi-dimensional binary classification problem, that is, finding an h-valued decision function $\boldsymbol{\phi}$ that maps every instance of n predictor variables $\mathbf{x}$ to a vector of h binary values $\mathbf{c} = (c_1, \ldots, c_h) \in \{-1, +1\}^h$:

$$\begin{aligned} \boldsymbol{\phi}: \quad \boldsymbol{\mathcal{X}} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \quad &\to \quad \{-1, +1\}^h \\ (x_1, \ldots, x_n) \quad &\mapsto \quad (c_1, \ldots, c_h), \end{aligned}$$

where $c_i = +1$ (-1) means that the i-th label is present (absent) in the predicted label subset Y. We consider the predictor variables $X_1, \ldots, X_n$ and the binary classes $C_i \in \{-1, +1\}$ as categorical random variables. Real examples include classification of texts into different categories [Gonçalves and Quaresma, 2003], diagnosis of multiple diseases from common symptoms and identification of multiple biological gene functions [Blockeel et al., 2006, Zhang and Zhou, 2007].

The easiest way to approach a multi-label classification problem is to divide it into a set of single-label classification problems (equivalent to binary classification problems). Each binary problem is then solved independently and thus h binary classifiers, one for each class variable C_i, are built. Each binary classifier is learned from predictor variables and C_i data only. At the end the results are combined to form multi-label prediction. Known as *binary relevance*, this method is easily implementable, has low computational

complexity and is fully parallelizable. Therefore it is scalable to a large number of classes. However, it completely ignores dependencies among labels and generally does not represent the most likely set of labels.

Chain classifiers [Read et al., 2009, Dembczynski et al., 2010] relax the independence assumption by iteratively adding class dependencies in the binary relevance scheme. The k-th classifier in the chain predicts class C_k from $X_1, \ldots, X_n, C_1, \ldots, C_{k-1}$. Sucar et al. [2014] employed naive Bayes within chain classifiers.

In this chapter, we study differences in the *expressive power* of these two methods when Bayesian network (BN) classifiers [Bielza and Larrañaga, 2014] are used. In particular, we extend the results of Chapter 3 to multi-label classifiers. Moreover, we suggest some theoretical reasons why the simple binary relevance method can perform poorly when relationships among labels exist, and we prove that chain classifiers provide more expressive models.

4.1.1 Chapter Outline

We describe the binary relevance method and compute its expressive power in Section 4.2. We analyze chain classifiers in Section 4.3. In Section 4.4 we compare the two methods, proving that actually chain classifiers are more expressive than binary relevance and in Section 4.6 we present our conclusions and some ideas for future research.

4.2 BAN Binary Relevance Classifiers

We consider the binary relevance method built upon BAN classifiers as base models, that is, for every class variable C_i we learn a BAN classifier with predictor subgraph $\mathcal{G}_i$. Thus we actually transform our multi-label problem into a number of single binary-class problems. The results of last chapter are then straightforwardly applied.

From Lemma 3.3 it follows that if $\boldsymbol{\phi} = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \ldots, \phi_h(\mathbf{x}))$ is the h-valued decision function induced by the h BAN classifiers, then there exist

$$p_1(\mathbf{x}) \in \mathcal{P}_{\mathcal{G}_1}, \ldots, p_h(\mathbf{x}) \in \mathcal{P}_{\mathcal{G}_h},$$

such that $\phi_k(\mathbf{x}) = sign(p_k(\mathbf{x}))$ for every $k \in \{1, \ldots, h\}$. We have then that the multi-valued decision function has a polynomial representation as,

$$\boldsymbol{\phi}(\mathbf{x}) = \big(sign(p_1(\mathbf{x})), \ldots, sign(p_h(\mathbf{x}))\big).$$

When we also assume that the predictor subgraphs $\mathcal{G}_1, \ldots, \mathcal{G}_h$ contain no V-structures, we have that, for every single binary-class problem, Theorem 3.3 applies. Thus, in Lemma 4.1, we bound the number of multi-valued decision functions representable by the BAN binary relevance method, when the predictor subgraphs $\{\mathcal{G}_k\}_{k=1}^h$ do not contain V-structures.

Lemma 4.1. *Consider h BAN classifiers to predict h binary classes. Suppose that the predictor subgraphs are $\mathcal{G}_1, \ldots, \mathcal{G}_h$ respectively and they contain no V-structures. We have that $N(\mathcal{G}_1, \ldots, \mathcal{G}_h)$, the number of h-valued decision functions representable by the BAN binary relevance method, satisfies*

$$N(\mathcal{G}_1, \ldots, \mathcal{G}_h) \leq \prod_{k=1}^{h} C(M, d_k),$$

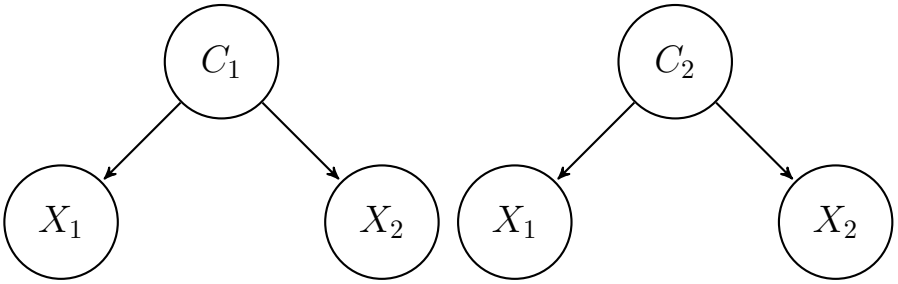


Figure 4.1: Two NB classifiers in Example 4.1

Table 4.1: Conditional probability tables in Example 4.1 for the NB of C_1

$\mathrm{P}(X_1\|C_1)$		X_1	
		0	1
C_1	-1	0.25	0.75
	$+1$	0.5	0.5

$\mathrm{P}(X_2\|C_1)$		X_2		
		2	3	4
C_1	-1	0.1	0.7	0.2
	$+1$	0.3	0.5	0.2

where $C(M,d) = 2\sum_{k=0}^{d-1}\binom{M-1}{k}$, $d_k = \sum_{i=1}^{n}\left((m_i-1)\prod_{s\in pa_k(i)} m_s\right)+1$, $pa_k(i)$ *is the set of* X_i *parents in* $\mathcal{G}_k$ *and* $M = \prod_{i=1}^{n} m_i$.

Proof. The proof is a straightforward application of Corollary 3.3. □

Remark 4.1. *We consider now, for visualization purposes, a simpler version of the above models. In particular when the predictors subgraphs are all the same, that is,* $\mathcal{G}_j = \mathcal{G}$. *The total number of* h*-valued decision functions over* n *categorical predictors is* $2^{h\prod m_i} = 2^{hM}$. *Then the fraction of* h*-valued decision functions representable by the BAN binary relevance method is bounded by*

$$\frac{N\left(\mathcal{G}_1,\ldots,\mathcal{G}_h\right)}{2^{hM}} \leq \left(\frac{C(M,d)}{2^M}\right)^h.$$

Thus, we have that if we fix the structure of the predictor subgraph, and it does not contain V-structures, the number of representable multi-valued decision functions becomes vanishingly small as the number of predictors increase. Moreover, using the binary relevance method, the speed at which the ratio between representable multi-valued decision functions and the total number of multi-valued decision functions drops to zero, is exponential in h*, the number of classes.*

Example 4.1. *We consider two binary classes* C_1, C_2 *and two predictor variables* $X_1 \in \{0,1\}$ *and* $X_2 \in \{2,3,4\}$. *Using the binary relevance method we build two independent NB classifiers, see Figure 4.1. Next, we list the conditional probability tables for both classifiers (Tables 4.1 and 4.2). Moreover, we consider uniform prior probabilities for both classes* C_1 *and* C_2.

From the representation of Theorem 3.3 we have that there exist two polynomials

Table 4.2: Conditional probability tables in Example 4.1 for the NB of C_2

$\mathrm{P}(X_1\|C_2)$		X_1	
		0	1
C_2	-1	0.4	0.6
	$+1$	0.7	0.3

$\mathrm{P}(X_2\|C_2)$		X_2		
		2	3	4
C_2	-1	0.6	0.2	0.2
	$+1$	0.1	0.1	0.8

p_1, p_2 that sign-represent the decision functions induced by the two NB classifiers

$$\begin{aligned} p_1(x_1,x_2) =& \log\left(\frac{0.5}{0.25}\right)\frac{x_1-1}{-1} + \log\left(\frac{0.5}{0.75}\right)\frac{x_1}{1} \\ &+ \log\left(\frac{0.3}{0.1}\right)\frac{(x_2-3)(x_2-4)}{2} + \log\left(\frac{0.5}{0.7}\right)\frac{(x_2-2)(x_2-4)}{-1} \\ &+ \log\left(\frac{0.2}{0.2}\right)\frac{(x_2-2)(x_2-3)}{2} \end{aligned}$$

and

$$\begin{aligned} p_2(x_1,x_2) =& \log\left(\frac{0.7}{0.4}\right)\frac{x_1-1}{-1} + \log\left(\frac{0.3}{0.6}\right)\frac{x_1}{1} \\ &+ \log\left(\frac{0.1}{0.6}\right)\frac{(x_2-3)(x_2-4)}{2} + \log\left(\frac{0.1}{0.2}\right)\frac{(x_2-2)(x_2-4)}{-1} \\ &+ \log\left(\frac{0.8}{0.2}\right)\frac{(x_2-2)(x_2-3)}{2}. \end{aligned}$$

We have that

$$\boldsymbol{\phi}(\mathbf{x}) = \Big(sign\big(p_1(\mathbf{x})\big), sign\big(p_2(\mathbf{x})\big)\Big)$$

is the bi-valued decision function that predicts C_1, C_2 from X_1, X_2. Figure 4.2 shows the decision boundaries of the two classifiers (black for C_1 and gray for C_2). We observe that the predictor space $\boldsymbol{\mathcal{X}} = \{0,1\} \times \{2,3,4\}$ is partitioned into four subsets corresponding to the four different predictions of the two binary classes. The value of the respective predicted class changes when one of the decision boundaries is crossed.

4.3 BAN Chain Classifiers

The easiest way to relax the strong independence assumption of the binary relevance method is to gradually add the predicted classes to the predictors. Specifically, suppose that we have to predict h binary classes $C_1, \ldots, C_h$ from n predictor variables $X_1, \ldots, X_n$. We consider h BAN classifiers such that the k-th BAN classifier predicts C_k from the variables

$$X_1, \ldots, X_n, C_1, \ldots, C_{k-1}.$$

In the predicting phase we will then use the predictor values and the previous predicted classes values $\hat{c}_1, \ldots, \hat{c}_{k-1}$ to predict class C_k. From Lemma 3.3 we have that there exist h polynomials $p_1, \ldots, p_h$

$$p_k(\mathbf{x}, \hat{c}_1, \ldots, \hat{c}_{k-1}) : \mathbb{R}^{n+k-1} \to \mathbb{R}$$

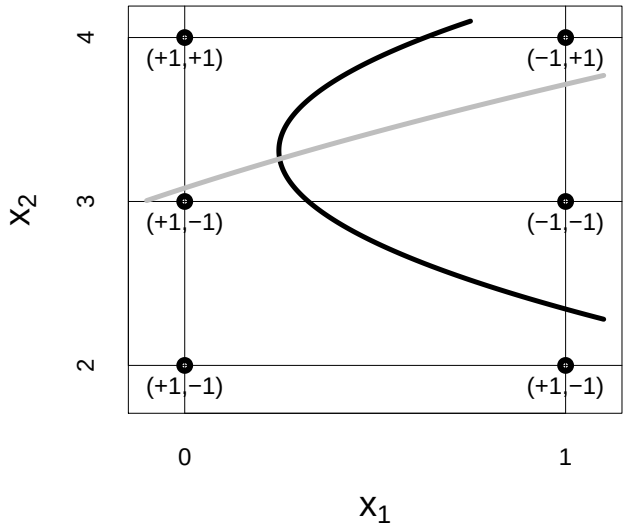


Figure 4.2: Decision boundaries for the two NB classifiers in Example 4.1, black for C_1 and gray for C_2. The value of the predicted classes is reported

$$p_k \in \mathcal{F}_{\mathcal{G}_k},$$

such that, if $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_h)$ is the multi-valued decision function associated with a chain classifier we have that,

$$\phi_k(\mathbf{x}) = sign(p_k(\mathbf{x}, \phi_1(\mathbf{x}), \ldots, \phi_{k-1}(\mathbf{x}))) \tag{4.1}$$

where $\mathcal{G}_k$ is the predictor subgraph related to the k-th BAN classifier over $X_1, \ldots, X_n$ and $C_1, \ldots, C_{k-1}$.

From now on we will focus on a particular and simpler form of BAN chain classifier, where the previous predicted classes are present in a naive way in the predictor subgraph. That is, $C_1, \ldots, C_{k-1}$ are not connected among them neither with other predictors in the subgraph $\mathcal{G}_k$. We refer to this kind of chain classifier as *naive BAN chain classifier*, see an example in Figure 4.3. As we will see those naive models have a more simpler representation of multi-valued decision functions and permit a deeper analysis. We observe that more complex chain models could be addressed in a similar way, using the interpolating polynomials to represent the decision functions of the already predicted classes. In more complex models, however, the analysis of the decision function is more difficult and not all the following results can be extended directly.

For a naive BAN chain classifier for $C_1, \ldots, C_h$, over $X_1, \ldots, X_n$ we denote by $\mathcal{H}_k$ the subgraph of the k-th BAN restricted to the original predictors $X_1, \ldots, X_n$.

Since classes C_j are binary, expanding Equation (4.1) we obtain the following sign-

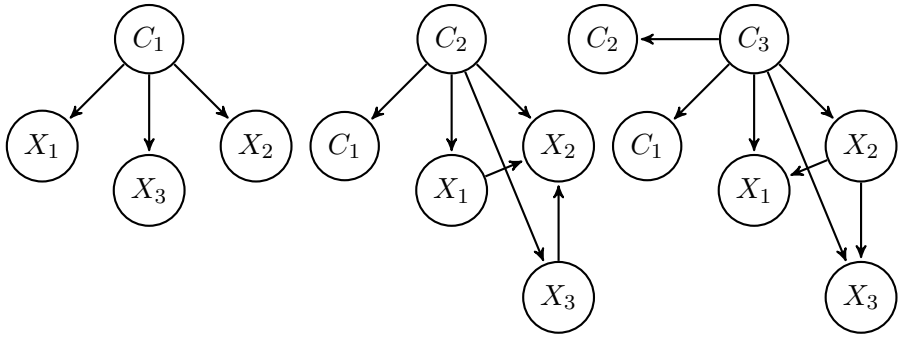


Figure 4.3: Example of naive BAN chain classifier with three classes and three predictor variables

representation of the k-th decision function in a naive BAN chain classifier:

$$\begin{aligned}
\phi_k(\mathbf{x}) &= sign(p_k(\mathbf{x}, \phi_1(\mathbf{x}), \ldots, \phi_{k-1}(\mathbf{x})) \\
&= sign\left(\sum_{i=1}^{n}\sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k}\in\mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s\in pa(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s) \right. \\
&\qquad \left. + \sum_{j=1}^{k-1}\left[\beta_j(-1)\ell_{-1}^{\{-1,+1\}}(\hat{c}_j) + \beta_j(+1)\ell_{+1}^{\{-1,+1\}}(\hat{c}_j)\right]\right) \\
&= sign\left(\hat{q}_k(\mathbf{x}) + \sum_{j=1}^{k-1}\left[\beta_j(-1)\ell_{-1}^{\{-1,+1\}}(\hat{c}_j) + \beta_j(+1)\ell_{+1}^{\{-1,+1\}}(\hat{c}_j)\right]\right),
\end{aligned}$$

where $\hat{q}_k \in \mathcal{P}_{\mathcal{H}_k}$, $\hat{c}_j = \phi_j(\mathbf{x})$ is the predicted value of the previous classifier expressed by the interpolating polynomial as a function of $\mathbf{x}$, $\ell_{-1}^{\{-1,+1\}}(c) = \frac{c-1}{-2}$ and $\ell_{+1}^{\{-1,+1\}}(c) = \frac{c+1}{2}$ are the Lagrange basis polynomials over $\{-1,+1\}$ and $\beta_j(c) = \log\left(\frac{\mathrm{P}(C_j=c|C_k=+1)}{\mathrm{P}(C_j=c|C_k=-1)}\right)$. Rearranging the terms in the sum we obtain that the following function sign-represents ϕ_k,

$$q_k(\mathbf{x}) = \hat{q}_k(\mathbf{x}) + \sum_{j=1}^{k-1}\left(a_j\phi_j(\mathbf{x}) + b_j\right), \tag{4.2}$$

where ϕ_j are the decision functions of the previous predicted class in the chain, $\hat{q}_k$ is the polynomial related to the subgraph $\mathcal{H}_k$ as in Theorem 3.3 and

$$a_j = \frac{1}{2}\log\left(\frac{\mathrm{P}(C_j=+1|C_k=+1)\,\mathrm{P}(C_j=-1|C_k=-1)}{\mathrm{P}(C_j=+1|C_k=-1)\,\mathrm{P}(C_j=-1|C_k=+1)}\right) \tag{4.3}$$

$$b_j = \frac{1}{2}\log\left(\frac{\mathrm{P}(C_j=+1|C_k=+1)\,\mathrm{P}(C_j=-1|C_k=+1)}{\mathrm{P}(C_j=+1|C_k=-1)\,\mathrm{P}(C_j=-1|C_k=-1)}\right) \tag{4.4}$$

Observe that we can omit constants b_j in Equation (4.2) if analysing the expressive power. In fact constants could be included in the polynomial $\hat{q}_k$ using elementary properties of Lagrange basis polynomials, see Chapter 3. The following lemma describes the set of decision functions induced by the k-th step of the naive BAN chain classifier.

Lemma 4.2. *Consider a multi-label classification problem over predictors $X_1, \ldots, X_n$ and a naive BAN chain classifier with predictor subgraphs $\mathcal{H}_1, \ldots, \mathcal{H}_h$ for classes ordered as $C_1, \ldots, C_h$. Assume that the predictor subgraphs do not contain V-structures. For every $k \in \{2, \ldots, h\}$ we have that, if $\phi_1, \ldots, \phi_{k-1}$ are the decision functions for $C_1, \ldots, C_{k-1}$ respectively, then the following set of polynomials sign-represent every decision function for class C_k,*

$$\mathcal{F}_{\mathcal{H}_k} + < \phi_1, \ldots, \phi_{k-1} >,$$

where $< \ldots >$ denotes the span of the included vectors and the sum is intended as the sum of two vectorial spaces, that is, the vectorial space which includes all the possible sum of elements of the two spaces, $\mathcal{F}_{\mathcal{H}_k}$ and $< \phi_1, \ldots, \phi_{k-1} >$.

Proof. The proof of the result is just an application of Theorem 3.3 and Equation (4.2). □

We have furthermore, that the set $sign\left(\mathcal{F}_{\mathcal{H}_k} + < \phi_1, \ldots, \phi_{k-1} >\right)$ is equal to the set of decision functions representable by the k-th BAN classifier of the naive BAN chain classifier if the graphs $\mathcal{H}_k$ do not contain V-structures. Intuitively, from an expressive-power point of view, we have the addition of the previous predicted classes in the k-th step of a naive BAN chain classifier being the equivalent to the *enrichment* of the space of functions $\mathcal{F}_{\mathcal{H}_k}$, related to the original predictors, by a subspace generated by the previously induced decision functions. To analyze if and how the enlarged space is indeed a bigger space, in other words, that it has a higher dimension, we have to understand when a decision function $\phi \in \mathcal{C}$ does not belong to a space of the type $\mathcal{F}_{\mathcal{G}}$ for some graph $\mathcal{G}$. Thus, in this case, adding $< \phi >$ to $\mathcal{F}_{\mathcal{G}}$ will actually increase the dimension.

First of all we define the set of relevant variables for a given decision function.

Definition 4.1. *Given a decision function*

$$\phi(x_1, \ldots, x_n) : \boldsymbol{\mathcal{X}} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \to \mathbb{R}$$

we say that a variable X_i is irrelevant for ϕ if

$$\phi(x_1, \ldots, x_n) = g(\mathbf{x}_{-i}) = \psi(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n), \quad \forall (x_1, \ldots, x_n) \in \boldsymbol{\mathcal{X}}.$$

A variable is said to be relevant for ϕ if it is not irrelevant, and we indicate with $\mathcal{V}(\phi)$ the set of relevant variables for ϕ.

As we will see relevant variables are important in order to determine if a given decision function belongs or not to some space $\mathcal{F}_{\mathcal{G}}$. In real applications the task of finding relevant variables of a decision function is computationally expensive and moreover in reality we usually know just an estimation of a decision function or its value on a set of random points. The presented analysis is thus intended as a theoretical analysis.

Example 4.2. *We show some examples of decision functions and their respective set of relevant variables.*

1. *If ϕ_1 is a decision function over $\{0, 1, 2\} \times \{-3, -2\}$, such that*

$$\phi_1(x_1, x_2) = \begin{cases} -1 & \text{if } (x_1, x_2) = (0, -3) \text{ or } (0, -2) \\ +1 & \text{otherwise.} \end{cases}$$

Then obviously $\phi_1(x_1, x_2) = g(x_1)$, where $g(x_1) = -1$ if $x_1 = 0$ and $+1$ otherwise. Thus X_2 is irrelevant for ϕ_1 and $\mathcal{V}(\phi_1) = \{X_1\}$.

2. *If ϕ_2 is the xor-function over $\{0,1\} \times \{0,1\}$, defined as follows*

$$\phi_2(x_1, x_2) = \begin{cases} -1 & \text{if } (x_1, x_2) = (0,0) \text{ or } (1,1) \\ +1 & \text{if } (x_1, x_2) = (0,1) \text{ or } (1,0). \end{cases}$$

Then $\mathcal{V}(\phi_2) = \{X_1, X_2\}$ and ϕ_2 does not have irrelevant variables.

3. *If ϕ_3 is the function over $\{0,1\} \times \{0,1\}$ such that,*

$$f_3(x_1, x_2) = \begin{cases} -1 & \text{if } (x_1, x_2) = (0,0) \\ +1 & \text{otherwise.} \end{cases}$$

Then also in this case $\mathcal{V}(\phi_3) = \{X_1, X_2\}$.

We can now state the following result about the set of relevant variables of decision functions.

Lemma 4.3. *Consider a graph $\mathcal{G}$ without V-structures and the space of functions $\mathcal{F}_\mathcal{G}$ defined in Equation (3.17). For every decision function ϕ we have that,*

$$\phi \in \mathcal{F}_\mathcal{G} \Leftrightarrow \mathcal{V}(\phi) \text{ are completely connected in } \mathcal{G}.$$

Proof. If the relevant variables for ϕ are completely connected in the graph $\mathcal{G}$, then we have that the polynomials in $\mathcal{F}_\mathcal{G}$ could interpolate over $\boldsymbol{\mathcal{X}}$ any function of variables in $\mathcal{V}(\phi)$ only. In particular, there exists a polynomial $p(\mathbf{x}) \in \phi_\mathcal{G}$ such that $\phi(\mathbf{x}) = p(\mathbf{x})$, $\forall \mathbf{x} \in \boldsymbol{\mathcal{X}}$ and thus $\phi \in \mathcal{F}_\mathcal{G}$.

To prove the other implication we observe that if two variables X_i and X_j are not directly connected in the graph $\mathcal{G}$, each polynomial $p(\mathbf{x}) \in \mathcal{F}_\mathcal{G}$ could be split into the sum of two polynomials,

$$p(\mathbf{x}) = p_1(\mathbf{x}_{-\{i,j\}}, x_i) + p_2(\mathbf{x}_{-\{i,j\}}, x_j). \tag{4.5}$$

To prove the above equality we just observe that each polynomial p in $\mathcal{F}_\mathcal{G}$ has the following expression

$$p(\mathbf{x}) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \ell_j^{\mathcal{X}_i}(x_i) \sum_{\mathbf{k} \in \mathbb{M}_i} \beta_i(j|\mathbf{k}) \prod_{s \in \mathbf{pa}(i)} \ell_{k_s}^{\mathcal{X}_s}(x_s).$$

Thus two variables appear in the same product of different Lagrange polynomial bases if and only if they are directly connected, that is, if and only if one variable belongs to the parents of the other. It is clear now that the sum in Equation (4.5) is therefore valid.

So we have only to prove that a decision function $\phi \in \mathcal{C}$ with two relevant variables $X_1 \in \mathcal{X}_1, X_2 \in \mathcal{X}_2$ could not be equal, over $\mathcal{X}_1 \times \mathcal{X}_2$, to the sum of two functions $p_1(x_1)$ and $p_2(x_2)$. Since X_1 and X_2 are relevant variables, there exist $s, s' \in \mathcal{X}_1$ and $t, t' \in \mathcal{X}_2$ such that,

$$\phi(s,t) = -\phi(s,t') \quad \text{and} \quad \phi(s,t) = -\phi(s',t)$$

Suppose $\phi(x_1, x_2) = p_1(x_1) + p_2(x_2)$, then we have,

$$\begin{aligned} \phi(s',t') =& p_1(s') + p_2(t') \\ =& p_1(s') + p_2(t) + p_1(s) + p_2(t') - p_1(s) - p_2(t) \\ =& \phi(s',t) + \phi(s,t') - \phi(s,t) = -3\phi(s,t). \end{aligned}$$

And we get $|\phi(s',t')| \neq 1$ which is absurd given that ϕ is a decision function ($|\phi(\mathbf{x})| = 1$). □

We return to points 2 and 3 of Example 4.2. In both cases functions ϕ_2 and ϕ_3 do not have irrelevant variables. Thus from Lemma 4.3 we have that $\phi_2, \phi_3 \notin \mathcal{F}_{NB}$. But $\phi_2 \notin sign(\mathcal{F}_{NB})$ (see the results of Ling and Zhang [2002]) while $\phi_3 \in sign(\mathcal{F}_{NB})$ (see proof of Theorem 4.1).

Thanks to Lemma 4.3, we have the following result.

Lemma 4.4. *Consider a multi-label classification problem over categorical predictors $X_1, \ldots, X_n$, for binary classes ordered as $C_1, \ldots, C_h$. Given a sequence of predictor subgraphs $\mathcal{H}_1, \ldots, \mathcal{H}_h$ without V-structures, let us consider $\phi = (\phi_1, \ldots, \phi_h)$ the h-valued decision functions of the corresponding naive BAN chain classifier. Then, for every $1 \leq k \leq h$, we have that*

$$|sign\,(\mathcal{P}_{\mathcal{H}_k} + < \phi_1, \ldots, \phi_{k-1} >)| \leq C(M, d_k + s) \leq C(M, d_k + k - 1),$$

where $M = |\mathcal{X}| = \prod_{i=1}^{n} m_i$, $d_k = \dim(\mathcal{P}_{\mathcal{H}_k})$, and s is equal to the number of functions among $\phi_1, \ldots, \phi_{k-1}$ such that their relevant variables are not completely connected in $\mathcal{H}_k$.

Proof. Suppose, $\phi_{i_1}, \ldots, \phi_{i_s}$ are the decision functions among $\phi_1, \ldots, \phi_{k-1}$ such that their relevant variables are not completely connected in $\mathcal{H}_k$. From Lemma 4.3 we have that,

$$\phi_{i_1}, \ldots, \phi_{i_s} \notin \mathcal{P}_{\mathcal{H}_k},$$

and that

$$\phi_i \in \mathcal{P}_{\mathcal{H}_k} \text{ for every } i \in \{1, \ldots, k-1\} \setminus \{i_1, \ldots, i_s\},$$

Thus we have

$$\mathcal{P}_{\mathcal{H}_k} + < \phi_1, \ldots, \phi_{k-1} > = \mathcal{P}_{\mathcal{H}_k} + < \phi_{i_1}, \ldots, \phi_{i_s} >,$$

and so

$$\dim\,(\mathcal{P}_{\mathcal{H}_k} + < \phi_1, \ldots, \phi_{k-1} >) \leq d_k + s \leq d_k + k - 1.$$

Analogously to Corollary 3.3 we have the corresponding bounding. □

Remark 4.2. *We observe that changing the order of classes in which the chain classifier is built implies a change in the expressive power of the resulting multi-label classifier. If the chain classifier is built with the class ordering $C_1, \ldots, C_h$, we have that the k-th classifier for C_k is more expressive than all the previous classifiers in the chain. In fact, from Equation (4.2), we have that if ϕ is a decision function representable by the j-th step of the chain classifier, then ϕ is representable by every successive steps of the chain classifier.*

Example 4.3. *We use a NB chain classifier over the prediction problems of Example 4.1. The NB classifier for predicting class C_1 is the same as in Example 4.1 (see Figure 4.1 left and Table 4.1). The predictors of the NB classifier for predicting C_2 now include C_1. We consider the same conditional probability tables as in Example 4.1 (Tables 4.1 and 4.2). Moreover we have to specify the conditional probabilities of C_1 given C_2 in the NB that predicts C_2. We set*

$$\mathrm{P}(C_1 = +1|C_2 = +1) = 0.3 \textit{ and } \mathrm{P}(C_1 = -1|C_2 = +1) = 0.7,$$

$$\mathrm{P}(C_1 = +1|C_2 = -1) = 0.9 \textit{ and } \mathrm{P}(C_1 = -1|C_2 = -1) = 0.1.$$

And, thus, coefficients a_1 and b_1 as defined in Equations (4.3) and (4.4) are given by

$$a_1 = \frac{1}{2}\log\left(\frac{0.3 \times 0.1}{0.9 \times 0.7}\right) \quad \textit{and} \quad b_1 = \frac{1}{2}\log\left(\frac{0.3 \times 0.7}{0.9 \times 0.1}\right).$$

We have that the decision function to predict C_2 is sign-represented by

$$q_2(x_1, x_2) = p_2(x_1, x_2) + a_1\phi_1(x_1, x_2) + b_1$$

where $\phi_1(x_1, x_2) = sign(p_1(x_1, x_2))$ and p_2 are defined in Example 4.1. The decision boundaries of the two classes are shown in Figure 4.4. We observe that the two boundaries are no longer independent; the decision boundary for the second class C_2 (dashed gray line) depends on the decision boundary of the first class C_1.

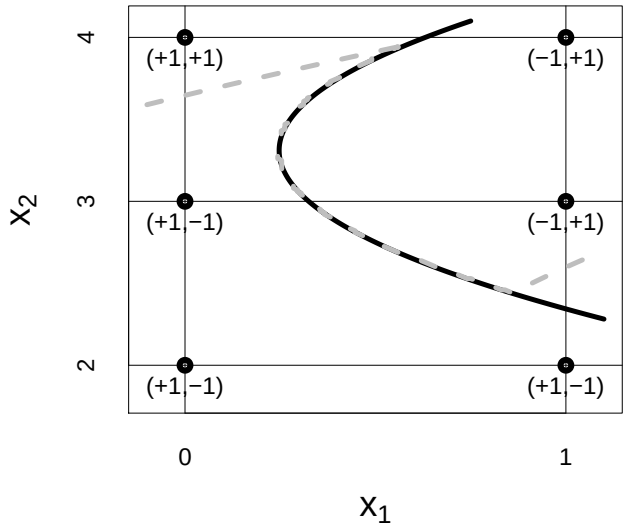


Figure 4.4: Decision boundaries for the chain NB classifier in Example 4.3. The value of the predicted classes is reported

4.3.1 Extensions to Classifier Trellises

Classifier trellises (CT) are a novel paradigm to multi-label classification problems, recently introduced by Read et al. [2015]. Basically CT works as chain classifiers, but instead of adding as predictors all the previous predicted classes, just some of them are considered in the new step of the classifier, thus reducing the complexity of the algorithm. We just observe here that our results about naive BAN chain classifiers could easily be extended to CT (when BAN classifiers are used as base models), especially when, as in naive BAN chain classifiers, the classes already predicted are added in a naive way.

4.4 Binary Relevance vs. Chain Classifiers

In this section, we compare the expressive power of binary relevance and chain classifiers when BAN classifiers are used as based models. We recall that a full Bayesian network is a Bayesian network where all pairs of nodes are linked.

Thanks to Lemma 4.3, we can prove the following result.

Theorem 4.1. *Consider a multi-label classification problem over categorical predictors $X_1 \in \mathcal{X}_1, \ldots, X_n \in \mathcal{X}_n$, for binary classes ordered as $C_1, \ldots, C_h$. Given a sequence of predictor subgraphs $\mathcal{H}_1, \ldots, \mathcal{H}_h$ without V-structures and such that they are not full Bayesian networks, consider $\mathcal{C}^h_{chain}$ to be the set of h-valued decision functions induced by the naive BAN chain classifier and $\mathcal{C}^h_{br}$ the set of h-valued decision functions induced by the corresponding binary relevance method. We have that,*

$$|\mathcal{C}^h_{chain}| > |\mathcal{C}^h_{br}|.$$

In other words, naive BAN chain classifiers are more expressive than the corresponding BAN binary relevance method.

Proof. From the results of the previous sections we have that,

$$\mathcal{C}^h_{br} = \{(\phi_1, \ldots, \phi_h) \text{ s.t. } \phi_k = sign(p_k),\, p_k \in \mathcal{F}_{\mathcal{H}_k}\}$$

$$\mathcal{C}^h_{chain} = \left\{ (\phi_1, \ldots, \phi_h) \text{ s.t. } \phi_k = sign\left(p_k + \sum_{j=1}^{k-1} a_j \phi_j \right),\, p_k \in \mathcal{P}_{\mathcal{H}_k},\, a_1, \ldots, a_{k-1} \in \mathbb{R} \right\}$$

Among the decision functions for the first class C_1 we can always choose for every $\mathbf{k} = (k_1, \ldots, k_n) \in \mathbb{M} = \{1, \ldots, m_1\} \times \cdots \times \{1, \ldots, m_n\}$, $\phi_{\mathbf{k}}(\mathbf{x})$ such that

$$\phi_{\mathbf{k}}(\mathbf{x}) = \begin{cases} +1 & \text{if } \mathbf{x} = (\xi_1^{k_1}, \ldots, \xi_n^{k_n}) \\ -1 & \text{if } \mathbf{x} \in \boldsymbol{\mathcal{X}} \setminus \{(\xi_1^{k_1}, \ldots, \xi_n^{k_n})\} \end{cases}$$

To prove the above fact is sufficient to observe that for every $\mathbf{k} \in \mathbb{M}$, $\phi_{\mathbf{k}}$ belongs to $sign(\mathcal{F}_{NB}) \subseteq sign(\mathcal{F}_{\mathcal{H}_1})$. In fact we have that $\phi_{\mathbf{k}} = sign(p(\mathbf{x}))$ where

$$\mathcal{F}_{NB} \ni p(\mathbf{x}) = \sum_{i=1}^{n} \ell_{k_i}^{\mathcal{X}_i}(x_i) - \sum_{i=1}^{n} \sum_{j \neq k_i} n\ell_j^{\mathcal{X}_i}(x_i),$$

as it is possible to check it by substitution.

Since $\mathcal{X}(\phi_{\mathbf{k}}) = \{X_1, \ldots, X_n\}$ and $\mathcal{H}_k$ is not complete, we have, from Lemma 4.3, that $\phi_{\mathbf{k}} \notin \mathcal{F}_{\mathcal{H}_k}$. Thus the space $\mathcal{F}_{\mathcal{H}_k} + < \phi_{\mathbf{k}} >$ has one dimension more than $\mathcal{F}_{\mathcal{H}_k}$, and so $sign(\mathcal{F}_{\mathcal{H}_k} + < \phi_{\mathbf{k}} >)$ contains at least two more decision functions than $sign(\mathcal{F}_{\mathcal{H}_k})$. So we have that there exist some h-valued decision functions that belong to $\mathcal{C}^h_{chain}$ but not to $\mathcal{C}^h_{br}$. □

We can also have a roughly estimation of the gain in expressibility from BAN binary relevance to naive BAN chain classifier.

Lemma 4.5. *If $\mathcal{C}^h_{chain}$ and $\mathcal{C}^h_{br}$ are defined as in Theorem 4.1 we have that*

$$|\mathcal{C}^h_{chain} \setminus \mathcal{C}^h_{br}| > |\mathcal{X}| \left(3^{h-1} - 1 \right).$$

Proof. As in the proof of Theorem 4.1 we can choose, among the decision functions for the first class C_1,

$$\phi_{\mathbf{k}}(\mathbf{x}) = \begin{cases} +1 & \text{if } \mathbf{x} = (\xi_1^{k_1}, \dots, \xi_n^{k_n}) \\ -1 & \text{if } \mathbf{x} \in \boldsymbol{\mathcal{X}} \setminus \{(\xi_1^{k_1}, \dots, \xi_n^{k_n})\} \end{cases}$$

Thus we have $|\boldsymbol{\mathcal{X}}|$ possibilities to choose the decision function for C_1. For every $\phi_{\mathbf{k}}$ we have two more decision functions representable for every other classes $C_2, \dots, C_k$, thus counting all the combinations we get

$$|\mathcal{C}^h_{chain} \setminus \mathcal{C}^h_{br}| > |\boldsymbol{\mathcal{X}}| \sum_{k=1}^{h-1} \binom{(h-1)}{k} 2^k = |\boldsymbol{\mathcal{X}}| \left(3^{h-1} - 1\right)$$

□

As we see from the proof, the estimation given by Lemma 4.5 is far from being sharp. However, it helps us to understand that chain classifiers are not just *more expressive* than binary relevance; the difference goes to $+\infty$ as the number of labels h grows.

4.5 Chain Regressors

Multi-output regression can be seen as the continuous alternative case to multi-label classification. The task is to predict the value of multiple continuous variables $Y_1, \dots, Y_h$ from a set of continuous predictors $X_1, \dots, X_n$. A review of methods can be found in Borchani et al. [2015].

Similarly to multi-label problems, two of the simplest algorithm are binary relevance (usually called *single target*, ST, in the multi-output regression context) and *chain regression* (CR). In particular chain regression, a problem transformation method, is directly inspired by the multi-label chain classifiers. Once an ordering of the output variables is chosen, they are predicted with single regression methods as in the ST method but in every step the k-th variable is estimated using the original predictors plus the previously predicted $k-1$ output variables. Obviously CR and ST methods can be used with whatever regression method as a base model.

Intuitively CR methods should exploit the possible relationship among output variables to deliver a better estimation, but actually in some cases building a chain regression is completely equivalent to the corresponding single target method.

This can be seen easily if linear regression is used as base model. The chain linear regression model consist in estimating Y_k with a linear regression over $X_1, \dots, X_n$ and $Y_1, \dots, Y_{k-1}$. Thus the estimator $\hat{Y}_k$ can be written iteratively as

$$\hat{Y}_k = \sum_{i=1}^{n} \beta_{k,i} X_i + \sum_{j=1}^{k-1} \beta_{k,n+j} \hat{Y}_j + \gamma_k \quad \forall k \in [h] \tag{4.6}$$

Obviously, since the system described by Equation (4.6) is triangular it is possible to express $\hat{Y}_k^{rc}$ with respect to the X_i only:

$$\hat{Y}_k = \sum_{i=1}^{n} \beta'_{k,i} X_i + \gamma'_k,$$

which is obviously a linear regression. Thus the use of a chain linear regression does not expand the expressive power of the model as in the multi-label case. In the multi-label setting a non-linear function (*sign*) is applied to the discrimination function.

Moreover we have that if ordinary least squares (OLS) or ridge regression [Hoerl and Kennard, 1970] is used to estimate the coefficients of the linear regressions, CR and ST methods yield exactly the same estimations of the coefficients as proved in the next lemma.

Lemma 4.6. *Chain linear regression is equivalent to single target linear regression if ordinary least squares or ridge estimators are used.*

Proof. Let $\mathbf{A}$ be the $N \times n$ matrix of input observation and $\mathbf{B}$ the $N \times h$ matrix of output observations, and assume $\mathbf{A}^t\mathbf{A}$ is invertible, otherwise OLS estimation cannot apply. Moreover we will denote with $\boldsymbol{\beta}_k \in \mathbb{R}^{n+k-1}$ the vectors of coefficients in the chain linear regression in Equation 4.6. Suppose the ordering of the chain is exactly $Y_1, \ldots, Y_h$. Then the coefficients of the first target are estimated as the OLS ones,

$$\mathbb{R}^m \ni \boldsymbol{\beta}_1 = \left(\mathbf{A}^t\mathbf{A}\right)^{-1}\mathbf{A}^t\mathbf{B}_1,$$

where $\mathbf{B}_1$ is the first column of $\mathbf{B}$, corresponding to the observations of Y_1. In the second training step of the chain, we compute the OLS estimation of the coefficients $\boldsymbol{\beta}_2$ of the regression of Y_2 over $X_1, \ldots, X_n, Y_1$. Thus:

$$\mathbb{R}^{n+1} \ni \boldsymbol{\beta}_2 = \left(\begin{array}{c|c} \boldsymbol{A}^t\boldsymbol{A} & \boldsymbol{A}^t\boldsymbol{B}_1 \\ \hline \boldsymbol{B}_1^t\boldsymbol{A} & \boldsymbol{B}_1^t\boldsymbol{B}_1 \end{array}\right)^{-1}\left(\begin{array}{c} \boldsymbol{A}^t \\ \hline \boldsymbol{B}_1^t \end{array}\right)\boldsymbol{B}_2.$$

Using the formula for computing the inverse of a block-defined matrix we obtain that

$$\left(\begin{array}{c|c} \boldsymbol{A}^t\boldsymbol{A} & \boldsymbol{A}^t\boldsymbol{B}_1 \\ \hline \boldsymbol{B}_1^t\boldsymbol{A} & \boldsymbol{B}_1^t\boldsymbol{B}_1 \end{array}\right)^{-1} = \left(\begin{array}{c|c} (\boldsymbol{A}^t\boldsymbol{A})^{-1} + \boldsymbol{\beta}_1\boldsymbol{C}\boldsymbol{D} & -\boldsymbol{\beta}_1\boldsymbol{C} \\ \hline -\boldsymbol{C}\boldsymbol{D} & \boldsymbol{C} \end{array}\right),$$

where

$$\begin{aligned} \boldsymbol{\beta}_1 =& (\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{B}_1 \in \mathbb{R}^{n\times 1}, \\ \boldsymbol{C} =& \left(\boldsymbol{B}_1^t\boldsymbol{B}_1 - \boldsymbol{B}_1^t\boldsymbol{A}(\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{B}_1\right)^{-1} \in \mathbb{R}^{1\times 1}, \\ \boldsymbol{D} =& \boldsymbol{\beta}_1^t = \boldsymbol{B}_1^t\boldsymbol{A}(\boldsymbol{A}^t\boldsymbol{A})^{-1} \in \mathbb{R}^{1\times n}. \end{aligned}$$

And we assume that $\left(\boldsymbol{B}_1^t\boldsymbol{B}_1 - \boldsymbol{B}_1^t\boldsymbol{A}(\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{B}_1\right)$ is invertible, that is, different from 0.

So we have that, splitting the vector of coefficients between the original predictors and the coefficient for Y_1, we obtain,

$$\boldsymbol{\beta}_2 = \left(\frac{\boldsymbol{\beta}_{2,1,\ldots,n}}{\beta_{2,n+1}}\right) = \left(\frac{(\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{B}_2 + \boldsymbol{\beta}_1\boldsymbol{C}\boldsymbol{D}\boldsymbol{A}^t\boldsymbol{B}_2 - \boldsymbol{\beta}_1\boldsymbol{C}\boldsymbol{B}_1^t\boldsymbol{B}_2}{-\boldsymbol{C}\boldsymbol{D}\boldsymbol{A}^t\boldsymbol{B}_2 + \boldsymbol{C}\boldsymbol{B}_1^t\boldsymbol{B}_2}\right).$$

And thus, the model of the first two step of the chain is,

$$\hat{y}_1 = \boldsymbol{\beta}_1^t\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad \text{and} \quad \hat{y}_2 = \boldsymbol{\beta}_2^t\begin{pmatrix} x_1 \\ \vdots \\ x_n \\ \hat{y_1} \end{pmatrix}.$$

Substituting now $\hat{y}_1$ into the equation for $\hat{y}_2$ we obtain that

$$\hat{y}_2 = \boldsymbol{\beta}^t_{2,1,\ldots,n} \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} + \beta_{2,n+1}\boldsymbol{\beta}^t_1 \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \left(\boldsymbol{\beta}^t_{2,1,\ldots,n} + \beta_{2,n+1}\boldsymbol{\beta}^t_1\right) \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}.$$

It is easy to see now by substitution that

$$\left(\boldsymbol{\beta}^t_{2,1,\ldots,n} + \beta_{2,n+1}\boldsymbol{\beta}^t_1\right) = (\boldsymbol{A}^t\boldsymbol{A})^{-1}\boldsymbol{A}^t\boldsymbol{B}_2. \tag{4.7}$$

The right-hand side of Equation (4.7) are the OLS estimations of the regression coefficients of Y_2 over $X_1, \ldots, X_n$. Hence the second step of the chain is equivalent to the OLS estimation of a ST model. Iterating the argument we obtain that every step of the chain is equivalent to the ST model. To prove the same statement for ridge regression estimation is sufficient to repeat the same argument used in the present proof using the ridge estimations of the parameters $(\boldsymbol{A}\boldsymbol{A}^t + \lambda\boldsymbol{I})^{-1}\boldsymbol{A}^t\boldsymbol{B}_1$ [Hoerl and Kennard, 1970]. □

4.6 Conclusions

In this chapter we have extended the results of Chapter 3 on the decision boundaries and expressive power of one-label BN classifiers to two types of BN multi-label classifiers: BAN classifiers built with binary relevance method and BAN chain classifiers. We have given theoretical grounds for why the binary relevance method provides models with poor expressive power and why this gets worst for larger number of classes. In both models, we have expressed the multi-label decision boundaries in polynomial forms and we have also proved that chain classifiers provide more expressive models than the binary relevance method when the same type of BAN classifier is used as base classifier.

Extending our results to general multi-dimensional BN classifiers [van der Gaag and de Waal, 2006, de Waal and van der Gaag, 2007, Bielza et al., 2011, Read et al., 2014], that permit BN structures within classes and predictors, is however, a much more complicated task. In multi-dimensional BN classifiers, the multi-valued decision functions have to be found by a global maximum search over the possible classes values. This fact does not permit the employment of the same arguments used in this work. It would be interesting to extend the *geometric* study of BAN classifiers, such as the study of the space of polynomials associated with every particular BAN. A deeper comprehension of the structure of $\mathcal{F}_{\mathcal{G}}$ could help to precisely compute or estimate the effective gain in expressive power of chain classifiers with respect to binary relevance when the same BAN classifiers are used as base model.

Chapter 5

Markov Property in Generative Classifiers

5.1 Introduction

Generative classifiers (see Section 2.3.3) are a wide class of machine learning models that consist of estimating the joint probability distributions over the predictor and class variables. From the estimated distribution a decision can be made over the class variable given the values of the predictors. It is well known that algebraic and geometric methods can be valuable tools in dealing with discrete probabilities as graphical models [Garcia et al., 2005, Settimi and Smith, 1998], contingency tables and exponential models [Diaconis and Sturmfels, 1995, Fienberg and Gilbert, 1970]. In this chapter we try to develop an algebraic and geometric point of view on generative binary classifiers over categorical predictors.

5.1.1 Chapter Outline

In Section 5.2 we introduce a discrete difference operator and we show its connection to conditional independence statement in generative classifiers. In Section 5.3 we study generative classifiers with undirected Markov property. We connect our findings with equalities of odds-ratios for multi-dimensional contingency tables in Section 5.4. In Section 5.5 we study maximum-likelihood estimation for parameters of generative classifiers, its limitations and an idea for combining the generative and discriminative approaches. Finally in Section 5.6 we resume the conclusion of the chapter.

5.2 Difference Operator and Conditional Independence

In this section we show that every conditional independence statement over the variables $(X_1, \ldots, X_n, C)$ is equivalent to a set of linear equations for the induced discrimination function, we then generalize the statement to undirected Markov networks. The result can be synthetically expressed using the difference operator centered in $\mathbf{x}^0 \in \boldsymbol{\mathcal{X}}$ and acting on any function $f : \boldsymbol{\mathcal{X}} \longrightarrow \mathbb{R}$.

Definition 5.1. *Let $f \in \mathcal{F}$ and $A \subseteq [n]$, the A-difference of first order (centered in*

$\mathbf{x}^0 \in \boldsymbol{\mathcal{X}}$) *is defined as,*

$$\Delta_A^{\mathbf{x}^0} f(\mathbf{x}) = f(\mathbf{x}) - f(\mathbf{x}_{-A}, \mathbf{x}_A^0).$$

Difference operators of order greater than one can be defined iteratively. In particular, for $A, B \subset [n]$ we are interested in the second order difference

$$\begin{aligned}\Delta_A^{\mathbf{x}^0} \Delta_B^{\mathbf{x}^0} f =& \Delta_A^{\mathbf{x}^0}(f(\mathbf{x}) - f(\mathbf{x}_{-B}, \mathbf{x}_B^0)) \\ =& f(\mathbf{x}) + f(\mathbf{x}_{-(A\cup B)}, \mathbf{x}_{A\cup B}^0) - f(\mathbf{x}_{-A}, \mathbf{x}_A^0) - f(\mathbf{x}_{-B}, \mathbf{x}_B^0).\end{aligned}$$

Lemma 5.1 connects the difference operators centered in different $\mathbf{x}^0, \mathbf{x}^1 \in \boldsymbol{\mathcal{X}}$.

Lemma 5.1. *For every $f \in \mathcal{F}$, $A, B \subseteq [n]$ and $\mathbf{x}^0, \mathbf{x}^1 \in \boldsymbol{\mathcal{X}}$.*

(i) $\Delta_A^{\mathbf{x}^1} f(\mathbf{x}) - \Delta_A^{\mathbf{x}^0} f(\mathbf{x}) = \Delta_A^{\mathbf{x}^1} f(\mathbf{x}_{-A}, \mathbf{x}_A^0)$

(ii) $\Delta_A^{\mathbf{x}^0} f(\mathbf{x}) = 0$ *for all* $\mathbf{x} \in \boldsymbol{\mathcal{X}}$ *if and only if* $\Delta_A^{\mathbf{x}^1} f(\mathbf{x}) = 0$ *for all* $\mathbf{x} \in \boldsymbol{\mathcal{X}}$

(iii) $\Delta_A^{\mathbf{x}^0} \Delta_B^{\mathbf{x}^0} f(\mathbf{x}) = 0$ *for all* $\mathbf{x} \in \boldsymbol{\mathcal{X}}$ *if and only if* $\Delta_A^{\mathbf{x}^1} \Delta_B^{\mathbf{x}^1} f(\mathbf{x}) = 0$ *for all* $\mathbf{x} \in \boldsymbol{\mathcal{X}}$.

Proof. For proving *(i)* we use Definition 5.1

$$\begin{aligned}\Delta_A^{\mathbf{x}^1} f(\mathbf{x}) - \Delta_A^{\mathbf{x}^0} f(\mathbf{x}) =& f(\mathbf{x}) - f(\mathbf{x}_{-A}, \mathbf{x}_A^1) - f(\mathbf{x}) + f(\mathbf{x}_{-A}, \mathbf{x}_A^0) \\ =& f(\mathbf{x}_{-A}, \mathbf{x}_A^0) - f(\mathbf{x}_{-A}, \mathbf{x}_A^1) = \Delta_A^{\mathbf{x}^1} f(\mathbf{x}_{-A}, \mathbf{x}_A^0).\end{aligned}$$

Points *(ii)* and *(iii)* follow now directly from *(i)*, we show this fact for point *(ii)*. Assume that $\Delta_A^{\mathbf{x}^1} f(\mathbf{x}) = 0$ for all $\mathbf{x} \in \boldsymbol{\mathcal{X}}$. From (i) we have that,

$$\Delta_A^{\mathbf{x}^0} f(\mathbf{x}) = \Delta_A^{\mathbf{x}^1} f(\mathbf{x}) - \Delta_A^{\mathbf{x}^1} f(\mathbf{x}_{-A}, \mathbf{x}_A^0).$$

Thus obviously $\Delta_A^{\mathbf{x}^0} f(\mathbf{x}) = 0$ for all $\mathbf{x} \in \boldsymbol{\mathcal{X}}$. Inverting now the roles of $\mathbf{x}^0$ and $\mathbf{x}^1$ in *(i)* we obtain the desired equivalence. □

Because of Lemma 5.1 we can assume $\mathbf{x}^0$ fixed and write Δ_A for $\Delta_A^{\mathbf{x}^0}$. Furthermore, if $f(\mathbf{x}) = 0$ for all $\mathbf{x} \in \boldsymbol{\mathcal{X}}$, we write $f \equiv 0$. The following lemma, whose proof follows directly from Definition 5.1, collects the basic properties of Δ_A.

Lemma 5.2. *Let $f, g \in \mathcal{F}$ and $A, B \subseteq [n]$.*

(i) $\Delta_A f \equiv 0$ *if and only if there exist a function h of the $\mathbf{x}_{-A}$ variables such that $f = h(\mathbf{x}_{-A})$,*

(ii) $f(\mathbf{x}) = h(\mathbf{x}_{-A}) + \Delta_A f(\mathbf{x})$,

(iii) $\Delta_A(\alpha f + \beta g) = \alpha \Delta_A f + \beta \Delta_A g$ *for all $\alpha, \beta \in \mathbb{R}$.*

And for the second order differences we can prove the following properties.

Lemma 5.3. *Let $f \in \mathcal{F}$ and $A, B \subseteq [n]$.*

(i) $\Delta_A \Delta_B f = \Delta_A f + \Delta_B f - \Delta_{A\cup B} f$.

(ii) $\Delta_A \Delta_A f = \Delta_A f$.

(iii) $\Delta_A\Delta_B f \equiv 0$ *if and only if there exist a function* h *of the* $\mathbf{x}_{-A}$ *variables and a function* g *of the* $\mathbf{x}_{-B}$ *variables such that* $f(\mathbf{x}) = h(\mathbf{x}_{-A}) + g(\mathbf{x}_{-B})$.

Proof. Points *(i)* and *(ii)* follow directly by Definition 5.1. To prove point *(iii)* we just observe that from point *(i)* of Lemma 5.2 we have that

$$\Delta_A\Delta_B f \equiv 0 \text{ if and only if } \Delta_B f = h(\mathbf{x}_{-A}),$$

and thus by point *(ii)* of Lemma 5.2, $f(\mathbf{x}) = g(\mathbf{x}_{-B}) + h(\mathbf{x}_{-A})$. □

The following observation gives an insight on why second order difference operators are meaningful to analyze conditional independence models.

Observation 5.1. *Consider* $\mathrm{P} \in \mathcal{P}$ *such that* $\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B | (\mathbf{X}_{-A\cup B}, C)$ *then we observe that the toric equation of the independence model (Proposition 3.1.4. in Drton et al. [2009]):*

$$\mathrm{P}(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_D, c)\,\mathrm{P}(\mathbf{x}'_A, \mathbf{x}'_B, \mathbf{x}_D, c) = \mathrm{P}(\mathbf{x}_A, \mathbf{x}'_B, \mathbf{x}_D, c)\,\mathrm{P}(\mathbf{x}'_A, \mathbf{x}_B, \mathbf{x}_D, c),$$

for all $c \in \{-1, +1\}$, $\mathbf{x}_D \in \boldsymbol{\mathcal{X}}_D$, $\mathbf{x}_B, \mathbf{x}'_B \in \boldsymbol{\mathcal{X}}_B$, $\mathbf{x}_A, \mathbf{x}'_A \in \boldsymbol{\mathcal{X}}_A$, *is equivalently written using the difference operator as*

$$\Delta_A\Delta_B \left(\log(\mathrm{P}(\mathbf{X} = \mathbf{x}, C = c))\right) = 0, \quad \forall c \in \{-1, +1\} \text{ and } \mathbf{x} \in \boldsymbol{\mathcal{X}}.$$

We can now prove that a conditional independence statement among the predictor variables is equivalent to the related second order difference of the discrimination function being equal to zero.

Lemma 5.4. *Let* $\mathbf{X} = (X_1, \ldots, X_n)$ *be a predictor vector of discrete random variables and* C *a binary class variable. Let* A, B, D *a partition of* $[n]$ *and* $f \in \mathcal{F}$. *The following statements are equivalent:*

(i) There exists a generative classifier $\mathrm{P} \in \Psi^{-1}(f)$ *such that* $\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B | (\mathbf{X}_D, C)$ *holds.*

(ii) $\Delta_A\Delta_B f \equiv 0$.

Proof. (i) $\Rightarrow$ *(ii)*: Let $\mathrm{P} \in \Psi^{-1}(f)$ be a probability distribution such that $\mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B | (\mathbf{X}_D, C)$. Thus $f(\mathbf{x}) = f_{\mathrm{P}}(\mathbf{x})$ and by Observation 5.1,

$$\Delta_A\Delta_B \left(\log(\mathrm{P}(\mathbf{X} = \mathbf{x}, C = c))\right) = 0, \quad \forall \mathbf{x} \in \mathbf{X} \text{ and } c \in \{-1, +1\}.$$

From the linearity of Δ_A (Lemma 5.2, *(iii)*) we have that $f_P = \log(\mathrm{P}(\mathbf{x}, +1)) - \log(\mathrm{P}(\mathbf{x}, -1))$

(ii) $\Rightarrow$ *(i)*:

We need to define P such that $\Delta_A\Delta_B \log(\mathrm{P}(\mathbf{X}, C)) \equiv 0$ and that $f = f_{\mathrm{P}}$. Given $\psi(\mathbf{x}) : \boldsymbol{\mathcal{X}} \mapsto \mathbb{R}$ such that $\Delta_A\Delta_B\psi \equiv 0$ (e.g., $\psi \equiv 0$), define

$$\log(\mathrm{P}(\mathbf{X} = \mathbf{x}, C = -1)) = \psi(\mathbf{x}) + k,$$
$$\log(\mathrm{P}(\mathbf{X} = \mathbf{x}, C = +1)) = \psi(\mathbf{x}) + k + f(\mathbf{x}),$$

where k is an appropriate normalization constant, that is,

$$\sum_{\mathbf{x}\in\boldsymbol{\mathcal{X}}} \exp(\psi(\mathbf{x})) \left(1 + e^{f(\mathbf{x})}\right) = \exp(-k).$$

P defined above obviously satisfies $\Delta_A\Delta_B \log(\mathrm{P}(\mathbf{X}, C)) \equiv 0$, moreover

$$f_{\mathrm{P}}(\mathbf{x}) = \psi(\mathbf{x}) + k + f(\mathbf{x}) - \psi(\mathbf{x}) - k = f(\mathbf{x}), \quad \forall \mathbf{x} \in \boldsymbol{\mathcal{X}}.$$

□

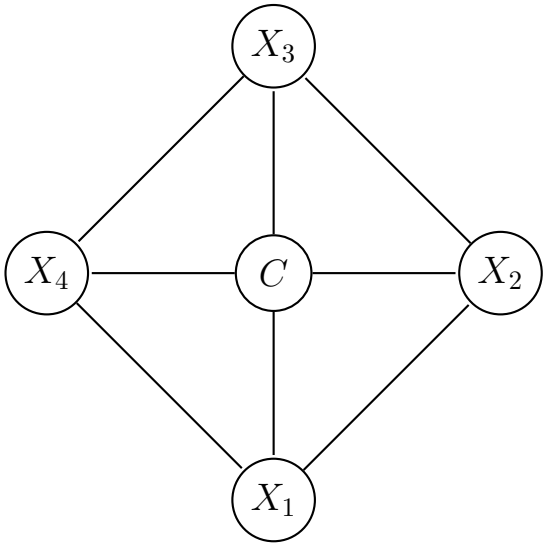


Figure 5.1: Markov classifier that is not equivalent to a BAN classifier

5.3 Markov network Classifiers

We consider now generative classifiers such that the underlying probability satisfies undirected Markov properties with respect to a given graph. In particular consider an undirected graph $\mathcal{G}$ over nodes indexed as variables $X_1, \ldots, X_n$, a $\mathcal{G}$-Markov classifier is defined as a generative classifier $\mathrm{P} \in \mathcal{P}$ such that both $\mathrm{P}(\mathbf{X}|C = +1)$ and $\mathrm{P}(\mathbf{X}|C = -1)$ satisfy the pairwise Markov property with respect to $\mathcal{G}$. Note that since generative classifiers are strictly positive probabilities, pairwise, local and global Markov properties are equivalent and Theorem 2.1 holds. Alternatively, we can define a $\mathcal{G}$-Markov classifier as a generative classifier P that satisfies pairwise (or equivalently global or local) Markov property with respect to an extended undirected graph; the extended graph is defined adding the node C to the graph $\mathcal{G}$ and connecting C to all predictor variables (See example in Figure 5.1).

For $\mathcal{G}$-Markov classifiers we can prove the following result.

Theorem 5.1. *The following statements are equivalent for every function $f : \boldsymbol{\mathcal{X}} \to \mathbb{R}$ and every undirect graph $\mathcal{G}$ over $X_1, \ldots, X_n$.*

(i) There exist a $\mathcal{G}$-Markov classifier $\mathrm{P} \in \Psi^{-1}(f)$.

(ii) $\Delta_A \Delta_B f \equiv 0$ for every A, B such that $\mathbf{X}_A$ and $\mathbf{X}_B$ are separated by the rest of variables in $\mathcal{G}$.

(iii) $f(\mathbf{x}) = \sum_{A \subseteq \{1,\ldots,n\}} \psi_A(\mathbf{x}_A)$, such that $\psi_A \equiv 0$ if $\mathbf{X}_A$ are not fully connected in $\mathcal{G}$.

Proof. (i) $\Rightarrow$ (ii): It is straightforward from Lemma 5.4.

(ii) $\Rightarrow$ (iii): As in the proof of Theorem 2.1 [Lauritzen, 1996], we consider

$$V_A(\mathbf{x}_A) = f(\mathbf{x}) - \Delta_A f(\mathbf{x}) = f(\mathbf{x}_{-A}, \mathbf{x}_A^0),$$

and

$$\psi_A(\mathbf{x}_A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} V_B(\mathbf{x}_B).$$

Thus from Möebius inversion lemma (Lemma 2.1) we have that

$$f(\mathbf{x}) = V_{[n]}(\mathbf{x}) = \sum_{A \subseteq [n]} \psi_A(\mathbf{x}_A).$$

We just have to show that $\psi_A \equiv 0$ if $\mathbf{X}_A$ are not completely connected in $\mathcal{G}$. Let $A \subset [n]$ such that $\mathbf{X}_A$ are not completely connected, then there exist $a, b \in A$ such that X_a and X_b are not adjacent, thus we can write, for $D = A \setminus \{a, b\}$,

$$\begin{aligned}\psi_A(\mathbf{x}) &= \sum_{B \subseteq D} (-1)^{|D \setminus B|} \left(V_B(\mathbf{x}_B) - V_{B \cup a}(\mathbf{x}_B, x_a) - V_{B \cup b}(\mathbf{x}_B, x_b) + V_{B \cup \{a,b\}}(\mathbf{x}_B, x_a, x_b) \right) \\ &= \sum_{B \subseteq D} (-1)^{|D \setminus B|} \Delta_a \Delta_b (f - \Delta_B f) = 0.\end{aligned}$$

Where the last equality is due to the linearity of the difference operator and the fact that $\Delta_a \Delta_b f = 0$ by point (ii) since a and b are not adjacent.

(iii) $\Rightarrow$ (i): We define a probability distribution P of the following form:

$$\log\left(\mathrm{P}(\mathbf{X} = \mathbf{x}, C = c)\right) = K + \sum_{A \subseteq [n]} \phi_A(\mathbf{x}_A, c), \tag{5.1}$$

where $\phi_A \equiv 0$ when $\mathbf{X}_A$ are not completely connected in $\mathcal{G}$. Using Theorem 2.1 we have that the conditional probabilities $\mathrm{P}(\mathbf{X}|C = \pm 1)$ are pairwise Markov with respect to $\mathcal{G}$. We choose now $\{\phi_A\}_{A \subseteq [n]}$ as

$$\phi_A(\mathbf{x}_A, c) = \begin{cases} g_A(\mathbf{x}_A) & \text{if } c = -1 \\ \psi_A(\mathbf{x}_A) + g_A(\mathbf{x}_A) & \text{if } c = +1 \end{cases} \tag{5.2}$$

where the $g_A(\mathbf{x}_A)$'s are arbitrary functions such that $g_A(\mathbf{x}_A) \equiv 0$ when $\mathbf{X}_A$ are not completely connected and K is the appropriate normalization factor. We obtain that the induced discrimination function $f_{\mathrm{P}} \equiv f$. Indeed

$$f_{\mathrm{P}} = \sum_{A \subseteq [n]} \left(\phi_A(\mathbf{x}_A, +1) - \phi_A(\mathbf{x}_A, -1) \right) = \sum_{A \subseteq [n]} \psi_A(\mathbf{x}_A).$$

□

When a function $f \in \mathcal{F}$ satisfies point *(ii)* for a given graph $\mathcal{G}$ we will concisely write $\Delta^2_{\mathcal{G}} f \equiv 0$. Moreover we can observe that the factorization in point *(iii)* can be concisely written using the set of cliques $\mathcal{K}(\mathcal{G})$ of the graph $\mathcal{G}$,

$$f(\mathbf{x}) = \sum_{A \in \mathcal{K}(\mathcal{G})} \psi_A(\mathbf{x}_A).$$

We show now that we can easily prove the following results equivalent to the result of Ling and Zhang [2002].

Corollary 5.1. *If $f \in \mathcal{F}$ is such that $\Delta^2_{\mathcal{G}} f \equiv 0$ and sign(f) contains a xor among variables $\mathbf{X}_A$, then $\mathbf{X}_A$ induce a complete subgraph in $\mathcal{G}$.*

Or equivalently if $\mathbf{X}_A$ is not completely connected in $\mathcal{G}$, it does not exist a $\mathcal{G}$-Markov classifier that can represent a xor among $\mathbf{X}_A$.

Proof. Let X_i and X_j non adjacent in $\mathcal{G}$ thus we have that $\Delta_i\Delta_j f \equiv 0$, and thus

$$f(\mathbf{x}) + f(\mathbf{x}_{-\{i,j\}}, x_i^0, x_j^0) = f(\mathbf{x}_{-i}, x_i^0) + f(\mathbf{x}_{-j}, x_j^0).$$

From the previous equation we see that it is impossible that

$$sign(f(\mathbf{x})) = sign(f(\mathbf{x}_{-\{i,j\}}, x_i^0, x_j^0)) = -sign(f(\mathbf{x}_{-i}, x_i^0)) = -sign(f(\mathbf{x}_{-j}, x_j^0)),$$

Since it is valid for every $\mathbf{x}, \mathbf{x}^0 \in \boldsymbol{\mathcal{X}}$ we have that no xor among X_i, X_j can be induced by f.

□

We can also prove a "relaxed" versions of the results in Theorem 5.1.

Corollary 5.2. *Given an undirected graph $\mathcal{G}$ we have that if $|\Delta^2_{\mathcal{G}} f| \leq \epsilon$*

$$f(\mathbf{x}) = \overline{f}(\mathbf{x}) + r(\mathbf{x}),$$

where $\Delta^2_{\mathcal{G}}\overline{f} \equiv 0$ and $|r(\mathbf{x})| \leq K\epsilon$, with K a constant that depends only on the graph $\mathcal{G}$. Vice versa if $f = \overline{f} + r$ with $|r| \leq \epsilon$ and $\Delta^2_{\mathcal{G}} \equiv 0$ then $|\Delta^2_{\mathcal{G}} f| \leq 4\epsilon$.

Proof. The proposition follows from the triangle inequality, the Möebius inversion formula as in the proof of Theorem 5.1 and from the linearity of Δ_A. □

5.3.1 Extended Markov Classifiers

In the same way extended Markov distributions are defined as limits of Markov distributions it is possible to define extended Markov classifiers.

Definition 5.2. P *is an extended $\mathcal{G}$-Markov classifier if there exist a sequence,* P^n *of $\mathcal{G}$-Markov classifiers that converges to* P*. That is,*

$$\mathrm{P}(\mathbf{X} = \mathbf{x}, C = c) = \lim_{n\to\infty} \mathrm{P}^n(\mathbf{X} = \mathbf{x}, C = c) \quad \forall \mathbf{x} \in \boldsymbol{\mathcal{X}}, c \in \{-1, +1\}.$$

We denote with $\overline{\mathcal{P}}(\mathcal{G})$ the set of extended $\mathcal{G}$-Markov classifiers. Observe that, in general, for an extended Markov classifier the induced discrimination function is not defined, simply because P does not have to be strictly positive. Nonetheless, the most-probable class and thus the induced decision function can be defined as

$$\phi_{\mathrm{P}}(\mathbf{x}) = \arg \max_{c\in\{-1,+1\}} \mathrm{P}(\mathbf{X} = \mathbf{x}, C = c).$$

If the graph $\mathcal{G}$ is the complete graph then we write simply $\overline{\mathcal{P}}$ for the set of extended generative classifiers.

In the next example we show how extended generative classifiers are connected with noise-free (or deterministic) classification problems.

Example 5.1. *Consider $f \in \mathcal{F}$ such that $\Delta^2_{\mathcal{G}} f \equiv 0$ for a given graph $\mathcal{G}$. And* P *a joint probability over* $\mathbf{X}$ *and C such that*

$$\mathrm{P}(C = c, \mathbf{X} = \mathbf{x}) = \begin{cases} 1 & \text{if } cf(\mathbf{x}) > 0 \\ 0 & \text{if } cf(\mathbf{x}) < 0 \end{cases}$$

P *represents a deterministic classification problem where* $\mathbf{X}$ *has a distribution given by* $\mathrm{P}(\mathbf{X})$ *and* C *is deterministically expressed by* $sign(f)$.

P *is obviously an extended generative classifier. In fact we have that*

$$\mathrm{P}^n(\mathbf{X}=\mathbf{x}, C=c) = \mathrm{P}(\mathbf{X}=\mathbf{x})\frac{\exp(ncf(\mathbf{x})/2)}{\exp(-nf(\mathbf{x})/2)+\exp(nf(\mathbf{x})/2)}$$

Converges to P *as* $n \to \infty$.

For a subset of $\overline{\mathcal{P}}(\mathcal{G})$ we can also extend the definition of induced discrimination function.

Definition 5.3. *We say that* P *is a marginally extended* $\mathcal{G}$*-Markov classifier if there exists* $\mathrm{P}^n \in \mathcal{P}_{\mathcal{G}} \cap \Psi^{-1}(f)$ *such that*

$$\mathrm{P}(\mathbf{X}=\mathbf{x}, C=c) = \lim_{n\to\infty} \mathrm{P}^n(\mathbf{X}=\mathbf{x}, C=c) \quad \forall \mathbf{x} \in \boldsymbol{\mathcal{X}}, c \in \{-1,+1\}.$$

We then denote with $f_{\mathrm{P}} = f$ *the induced discrimination function as for generative classifiers.*

With $\overline{\mathcal{P}}_{\mathcal{G}}(f)$ we denote the marginally extended Markov classifier with discrimination function f.

Observe that the induced discrimination function f_{P} for marginally extended Markov models is well-defined over the $\mathbf{x} \in \boldsymbol{\mathcal{X}}$ such that $\mathrm{P}(\mathbf{x}) > 0$.

5.3.2 Gaussian Predictors

In this section we show it is possible to prove a similar result to Theorem 5.1 for Gaussian predictor variables, namely that every discrimination functions in a given class can be induced by Markov classifiers with a given graph. We consider now the binary classification problem over continuous predictors, we assume moreover that the random vector of predictors $\mathbf{X}$ follows a normal distribution conditioned to the value of the class variable C. Namely we have that

$$\mathbf{X}|(C=c) \sim \mathcal{N}(\boldsymbol{\mu}_c, \Sigma_c).$$

Define P to be the probability distribution over $\mathbf{X}$ and C obtained by the previous equation and by the class prior $\mathrm{P}(C=+1) = 1-(C=-1) = p_+$. We call the probabilistic classifier obtained in this way a Gaussian classifier.

Given an undirected graph $\mathcal{G}$ with nodes indexed as the predictor variables $X_1, \ldots, X_n$ we say that a Gaussian classifier is Markov with respect to $\mathcal{G}$ if both the distributions $\mathrm{P}(\mathbf{X}|C=\pm 1)$ are Markov with respect to $\mathcal{G}$, that is both the concentration matrix $(\Sigma_{+1})^{-1}$ and $(\Sigma_{-1})^{-1}$ are Markov with respect to $\mathcal{G}$.

Definition 5.4. *A simmetric matrix* $A \in \mathbb{R}^{n\times n}$ *is said to be Markov with respect to an undirect graph* $\mathcal{G}$ *if and only if* $A_{i,j} = A_{j,i} = 0$ *for every* (i,j) *such that the* i*-th node is not adjacent to the* j*-th node.*

The Gaussian classifier is well studied[Duda et al., 2000] and the corresponding discrimination function is

$$f_{\mathrm{P}}(\mathbf{x}) = -\frac{1}{2}\mathbf{x}^t A \mathbf{x} + \alpha^t \mathbf{x} + \gamma, \tag{5.3}$$

where $A = \left(\Sigma_{+1}^{-1} - \Sigma_{-1}^{-1}\right)$, $\boldsymbol{\alpha}^t = \boldsymbol{\mu}_{+1}^t \Sigma_{+1}^{-1} - \boldsymbol{\mu}_{-1}^t \Sigma_{-1}^{-1}$ and $\gamma = \log\left(\frac{p_+}{1-p+}\right) - \frac{1}{2}\log\left(\frac{|\Sigma_{+1}|}{|\Sigma_{-1}|}\right) - \frac{1}{2}\left(\boldsymbol{\mu}_{+1}^t \Sigma_{+1}^{-1} \boldsymbol{\mu}_{+1} - \boldsymbol{\mu}_{-1}^t \Sigma_{-1}^{-1} \boldsymbol{\mu}_{-1}\right)$. The discriminant function is linear if the covariance matrices are equal give the values of the class variable, otherwise the discriminant function is quadratic in the predictor variables. In the case of markov Gaussian model Lemma 5.5 holds, similarly to the discrete case.

Lemma 5.5. *The following are equivalent for every undirect graph $\mathcal{G}$ and every function f*

(i) $f(\mathbf{x}) = -\frac{1}{2}\mathbf{x}^t A \mathbf{x} + \alpha^t \mathbf{x} + \gamma$ *with A simmetric and Markov with respect to $\mathcal{G}$,*

(ii) f *is induced by a Gaussian classifier Markov with respect to $\mathcal{G}$.*

Proof. (i) $\Rightarrow$ (ii): To prove the implication lets define a Gaussian classifier which is Markov with respect to $\mathcal{G}$ and that induces f. A Gaussian classifier is defined by the following parameters

- $p_+ = \mathrm{P}(C = +1)$,
- Σ_{+1} and Σ_{-1}, the covariance matrices,
- μ_{+1} and μ_{-1} the mean vectors.

We start by defining the covariance matrices as follow

$$\Sigma_{+1} = (A + \lambda S)^{-1},$$
$$\Sigma_{-1} = (\lambda S)^{-1}.$$

where $S \in \mathbb{R}^{n \times n}$ is any positive definite symmetric matrix Markov with respect to $\mathcal{G}$ and λ is an appropriate positive number. With this choice we have that:

$$(\Sigma_{+1})^{-1} - (\Sigma_{-1})^{-1} = A.$$

We just have to show that we can choose λ such that $(A + \lambda S)$ is positive definite (it is obviously symmetric and Markov with respect to $\mathcal{G}$). We have that:

$$\det\left(S + \frac{1}{\lambda}A\right) = \det(S) + \frac{1}{\lambda}\det(S)\,\mathrm{tr}(S^{-1}A) + \mathcal{O}\left(\frac{1}{\lambda^2}\right)$$

Thus there exist λ^0 such that $\det\left(S + \frac{1}{\lambda}A\right) > 0$ if $\lambda > \lambda^0$. If s_* is the smallest eigenvalue of S we have that $\mathbf{x}^t S \mathbf{x} \geq s_* \mathbf{x}^t \mathbf{x}$ for every $\mathbf{x}$. Choose now $\lambda > \frac{||A||_2}{s_*}$ we have that for every non-zero vector $\mathbf{x}$,

$$\mathbf{x}^t(\lambda S + A)x = \lambda \mathbf{x}^t S \mathbf{x} + \mathbf{x}^t A \mathbf{x} > \mathbf{x}^t \mathbf{x}\,(||A||_2 - ||A||_2) = 0.$$

Finally we have that $\lambda S + A$ is positive definite if we choose $\lambda > \max\left\{\lambda^0, \frac{||A||_2}{s_*}\right\}$.

We can now easily pick p_+ and the mean vectors to adjust the remaining two terms of f ($\boldsymbol{\alpha}$ and γ). In particular for $\boldsymbol{\alpha}$, it is sufficient to choose $\boldsymbol{\mu}_-$ equal to the zero vector and $\boldsymbol{\mu}_+ = (\Sigma_{+1})\boldsymbol{\alpha}$.

(ii) $\Rightarrow$ (i) is obvious from Equation 5.3. □

5.4 Constant Interactions Models

In this section we study the set of generative classifiers such that their discrimination function factorizes as in Theorem 5.1 for a given undirected graph. In particular if $\mathcal{G}$ is an undirected graph we are here interested in the following set,

$$\Psi^{-1}(\{f \text{ s.t } \Delta^2_{\mathcal{G}} f \equiv 0\}) = \{\mathrm{P} \in \mathcal{P} \text{ s.t. } \Delta^2_{\mathcal{G}} f_{\mathrm{P}} \equiv 0\} = \{\mathrm{P} \in \mathcal{P} \text{ s.t. } f_{\mathrm{P}}(\mathbf{x}) = \sum_{A \in \mathcal{K}(\mathcal{G})} f_A(\mathbf{x}_A)\}.$$

We prove that $\Delta_A \Delta_B f_{\mathrm{P}} \equiv 0$ can be stated as an equivalence among odds ratios of the contingency tables for the conditional probabilities given the class values. We will first show it in the simplest model with two binary predictors as it is linked with the well studied geometry of 2×2 contingency tables [Fienberg and Gilbert, 1970]. Then we will extend it to the general case.

2×2 Predictors

Consider the space of generative classifiers over two binary predictor variables with $X_1, X_2 \in \{0, 1\}$. The only not naive factorization of the discrimination function, is in this case, the one induced by the graph with no arcs among predictors. We are thus interested in

$$\{\mathrm{P}(X_1, X_2, C) \in \mathcal{P} \text{ s.t. } \Delta_1 \Delta_2 f_{\mathrm{P}} \equiv 0\}.$$

Since the predictor variables are binary, $\Delta_1 \Delta_2 f_{\mathrm{P}} \equiv 0$ reduces to only one equation,

$$f_{\mathrm{P}}(0,0) + f_{\mathrm{P}}(1,1) = f_{\mathrm{P}}(0,1) + f_{\mathrm{P}}(1,0).$$

By the definition of discrimination function (Definition 2.3) and the strict positivity of generative classifier probabilities, the above identity is equivalent to,

$$\frac{\mathrm{P}(0,0|C=+1)\,\mathrm{P}(1,1|C=+1)}{\mathrm{P}(0,1|C=+1)\,\mathrm{P}(1,0|C=+1)} = \frac{\mathrm{P}(0,0|C=-1)\,\mathrm{P}(1,1|C=-1)}{\mathrm{P}(0,1|C=-1)\,\mathrm{P}(1,0|C=-1)}. \tag{5.4}$$

The left and right hand side of Equation (5.4) are the odds ratios [Fienberg, 1968, Carlini and Rapallo, 2005] of respectively $\mathrm{P}(X_1, X_2|C=+1)$ and $\mathrm{P}(X_1, X_2|C=-1)$. We have thus that $\mathrm{P} \in \Psi^{-1}(\{f \text{ s.t. } \Delta_1 \Delta_2 f \equiv 0\})$ if and only if $\mathrm{P}(X_1, X_2|C=+1)$ and $\mathrm{P}(X_1, X_2|C=-1)$ have the same odds ratio.

Let us fix a positive real number $\alpha > 0$ and define the space of probability distributions over X_1, X_2 with fixed odds ratios equals to α as,

$$\mathcal{M}(\alpha) = \left\{ Q(X_1, X_2) \text{ s.t. } \frac{Q(0,0)Q(1,1)}{Q(0,1)Q(1,0)} = \alpha \right\}.$$

Obviously the set $\mathcal{M}(\alpha)$ is not empty for every $\alpha \in \mathbb{R}$. Moreover the set $\mathcal{M}(1)$ is the manifold of independent probabilities.

General Predictors

To extend the previous observation to general Markov classifiers, we need to define odds ratios for general models of more than two and not only binary, random variables. The following definition is a simple extension to multivariate tables of the odds ratios for $r \times c$ contingency tables in Fienberg [1968].

Definition 5.5. *Let A, B be two disjoint subsets of $[n]$, we define the (A, B) odds ratios of a probability Q over $\boldsymbol{\mathcal{X}}$ as*

$$\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D) = \frac{\mathrm{Q}(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_D)\,\mathrm{Q}(\mathbf{x}'_A, \mathbf{x}'_B, \mathbf{x}_D)}{\mathrm{Q}(\mathbf{x}_A, \mathbf{x}'_B, \mathbf{x}_D)\,\mathrm{Q}(\mathbf{x}'_A, \mathbf{x}_B, \mathbf{x}_D)}$$

where $D = [n] \setminus (A \cup B)$.

The set of all the, $|\boldsymbol{\mathcal{X}}||\boldsymbol{\mathcal{X}}_A||\boldsymbol{\mathcal{X}}_B|$, odds ratios in Definition 5.5 forms the *complete set* of (A, B) odds ratios.

The $(A.B)$ odds ratios satisfy the following properties for every probability Q, every disjoint sets A, B and every $\mathbf{x}_A, \mathbf{x}'_A, \mathbf{x}''_A \in \boldsymbol{\mathcal{X}}_A$, $\mathbf{x}_B, \mathbf{x}'_B, \mathbf{x}''_B \in \boldsymbol{\mathcal{X}}_B$, $\mathbf{x}_D \in \boldsymbol{\mathcal{X}}_D$.

$$\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D) = \alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}_B; \mathbf{x}_D) = 1$$
$$\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D) = \big(\alpha_{A,B}[\mathrm{Q}](\mathbf{x}'_A, \mathbf{x}_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D)\big)^{-1}$$
$$\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D) = \big(\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}'_B, \mathbf{x}_B; \mathbf{x}_D)\big)^{-1}$$
$$\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D)\alpha_{A,B}[\mathrm{Q}](\mathbf{x}''_A, \mathbf{x}_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D) = \alpha_{A,B}[Q](\mathbf{x}''_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D)$$
$$\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}_B, \mathbf{x}'_B; \mathbf{x}_D)\alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}''_B, \mathbf{x}_B; \mathbf{x}_D) = \alpha_{A,B}[\mathrm{Q}](\mathbf{x}_A, \mathbf{x}'_A; \mathbf{x}''_B, \mathbf{x}'_B; \mathbf{x}_D)$$

From the above equations we can see that the complete set of (A, B) odds ratios is not composed by independent values. It is known [Fienberg, 1968] that we can choose among the $|\boldsymbol{\mathcal{X}}||\boldsymbol{\mathcal{X}}_A||\boldsymbol{\mathcal{X}}_B|$ odds ratios a subset of $(|\boldsymbol{\mathcal{X}}_A|-1)(|\boldsymbol{\mathcal{X}}_B|-1)|\boldsymbol{\mathcal{X}}_D|$ elements that completely describe the complete set of odds ratios. One way to choose such a restricted subset is given by the *spanning cells odds ratios* [Fienberg, 1968] centered in a given point $(\mathbf{x}^0_A \in \boldsymbol{\mathcal{X}}_A, \mathbf{x}^0_B \in \boldsymbol{\mathcal{X}}_B)$.

Definition 5.6. *The (A, B) spanning cells odds ratios centered in $(\mathbf{x}^0_A \in \boldsymbol{\mathcal{X}}_A, \mathbf{x}^0_B \in \boldsymbol{\mathcal{X}}_B)$ are*

$$\alpha^{\mathbf{x}^0_A, \mathbf{x}^0_B}_{A,B}[\mathrm{Q}](\mathbf{x}) = \alpha_{A,B}[\mathrm{Q}](\mathbf{x}^0_A, \mathbf{x}_A; \mathbf{x}^0_B, \mathbf{x}_B; \mathbf{x}_D)$$

The (A, B) spanning cell odds ratios satisfy the following constrains:

$$\alpha^{\mathbf{x}^0_A, \mathbf{x}^0_B}_{A,B}[\mathrm{Q}](\mathbf{x}) = 1 \quad \text{if} \quad \mathbf{x}_A = \mathbf{x}^0_A \text{ or } \mathbf{x}_B = \mathbf{x}^0_B. \tag{5.5}$$

The spanning cell odds ratios thus consist of $(|\boldsymbol{\mathcal{X}}_A| - 1)(|\boldsymbol{\mathcal{X}}_B| - 1)|\boldsymbol{\mathcal{X}}_D|$ independent positive numbers.

We will call a function $\alpha : \boldsymbol{\mathcal{X}} \mapsto \mathbb{R}_{>0}$ that satisfies Equation (5.5) an (A, B) spanning cell odds ratio function (centered in $(\mathbf{x}^0_A, \mathbf{x}^0_B)$). We can thus define, for every (A, B) spanning cell odds ratio function α the *set of constant (A, B) interactions* as

$$\mathcal{M}_{A,B}(\alpha) = \left\{ \mathrm{Q} \text{ p.d.f. over } \boldsymbol{\mathcal{X}} \text{ s.t. } \alpha^{\mathbf{x}^0_A, \mathbf{x}^0_B}_{A,B}[\mathrm{Q}] \equiv \alpha \right\}.$$

The following representation holds for the set $\mathcal{M}(\alpha)$.

Lemma 5.6. *For every α an (A, B) spanning cell odds ratios function centered in $\mathbf{x}^0_A, \mathbf{x}^0_B$ we have that*

$$\mathcal{M}_{A,B}(\alpha) \simeq \frac{S_{A,B}(\alpha)}{\sim}.$$

Where $S_{A,B}(\alpha)$ is the linear space of the solutions of the following linear system over $\{l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D}\}_{\mathbf{x}\in\boldsymbol{\mathcal{X}}}$

$$l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} + l_{\mathbf{x}^0_A,\mathbf{x}^0_B,\mathbf{x}_D} - l_{\mathbf{x}^0_A,\mathbf{x}_B,\mathbf{x}_D} - l_{\mathbf{x}_A,\mathbf{x}^0_B,\mathbf{x}_D} = \log(\alpha(\mathbf{x})) \quad \forall \mathbf{x} \in \boldsymbol{\mathcal{X}}. \tag{5.6}$$

And $\sim$ is the equivalence relationship defined by

$$\{l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D}\}_{\mathbf{x}\in\boldsymbol{\mathcal{X}}} \sim \{l'_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D}\}_{\mathbf{x}\in\boldsymbol{\mathcal{X}}} \quad \Leftrightarrow \quad l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} - l'_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} = k \quad \forall \mathbf{x} \in \boldsymbol{\mathcal{X}}$$

Proof. Consider Q a p.d.f. over $\boldsymbol{\mathcal{X}}$ and define $l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} = \log(\mathrm{Q}(\mathbf{x}))$, we thus have that $\{l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D}\}_{\mathbf{x}\in\boldsymbol{\mathcal{X}}}$ satisfy the linear system in Eq. (5.6) if and only if $\mathrm{Q} \in \mathcal{M}_{A,B}(\alpha)$.

Eq. (5.6) can be solved for $l_{\mathbf{x}}$ and thus the space of solutions $S_{A,B}$ is not empty and has dimension equal to $(|\boldsymbol{\mathcal{X}}_A| + |\boldsymbol{\mathcal{X}}_B| - 1)|\boldsymbol{\mathcal{X}}_D|$.

We observe that every constant $l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} = k \in \mathbb{R}$ is a solution of the homogeneous system associate with Eq. (5.6) thus we have that for every $l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} \in S_{A,B}$ we can associate the following p.d.f.

$$\mathrm{Q}(\mathbf{x}) = \frac{\exp(l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D})}{\sum_{\mathbf{x}} \exp(l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D})} = \exp\left(l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} - \log\left(\sum_{\mathbf{x}} \exp(l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D})\right)\right).$$

Where $l^*_{\mathbf{x}} = l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D} - \log\left(\sum_{\mathbf{x}} \exp(l_{\mathbf{x}_A,\mathbf{x}_B,\mathbf{x}_D})\right)$ belong to $S_{A,B}$ being the sum of a solution of the linear system and a solution of the associate homogeneous system. Using $l^*_{\mathbf{x}}$ as the representative of the $\sim$-equivalence class of $l_{\mathbf{x}}$ it is clear that $\mathcal{M}_{A,B}(\alpha) \simeq \frac{S_{A,B}(\alpha)}{\sim}$ through the component-wise exponential map. □

As in the 2×2 case, we have that the manifold of independence models is obtained setting α as the (A, B) spanning cell odds ratio function constant 1. Thus $\mathcal{M}_{A,B}(1) = \{\mathrm{Q}(\mathbf{X}_A, \mathbf{X}_B) \text{ s.t. } \mathbf{X}_A \perp\!\!\!\perp \mathbf{X}_B\}$. Moreover the follow characterization holds.

Proposition 5.1. *For every two disjoint $A, B \subset [n]$ we have that $\Delta_A \Delta_B f_{\mathrm{P}} \equiv 0$ if and only if $\alpha_{A,B}[\mathrm{P}(\mathbf{x}|C = +1)] \equiv \alpha_{A,B}[\mathrm{P}(\mathbf{x}|C = -1)]$.*

Thus we have that for an undirected graph $\mathcal{G}$ the set of discrimination function representable by $\mathcal{G}$-Markov classifiers, that is $\{f \text{ s.t. } \Delta^2_{\mathcal{G}} \equiv 0\}$, is identical to the set of discrimination functions representable by all generative classifiers such that $\alpha_{A,B}[\mathrm{P}(\mathbf{X}|C = +1)] = \alpha_{A,B}[\mathrm{P}(\mathbf{X}|C = -1)]$ for every A, B separated by the rest of variables in $\mathcal{G}$. Observe that this last set of generative classifiers includes the $\mathcal{G}$-Markov classifiers but many more.

5.5 Parameters Estimation

In this section we study estimation methods for generative classifiers, that is, methods to fit the probability distributions. The method generally used to fit the parameters of a generative classifier is the maximum-likelihood estimation. It is well-known that, if the model is misspecified, the maximum-likelihood estimation is not consistent [Devroye et al., 1996], that is is not able to learn a model that induces the Bayes classifier even with an infinite amount of data. As we show in the next section even if the *real and unknown* decision function belongs to the set of decision functions that are representable by $\mathcal{G}$-Markov classifiers, if the probability does not satisfies the pairwise Markov property with respect to $\mathcal{G}$ than the maximum-likelihood estimation is not optimal. To show this fact is sufficient to study the naive Bayes model.

5.5.1 Non Optimality

Domingos and Pazzani [1997] observe that the naive Bayes classifier is not optimal under 0-1 loss if the conditional probabilities $\mathrm{P}(X_i|C)$ are estimated from data with maximum-likelihood (empirical frequencies), even if the real decision function f is linear (that is $\Delta_i\Delta_j f \equiv 0$ in our framework).

The example cited by Domingos and Pazzani [1997] are the so-called m-of-n concepts. An m-of-n concept is a classification problem, where the $C = +1$ if at least m of the n binary predictors ($X_i \in \{0, 1\}$) are 1 and $C = -1$ otherwise. Observe that a m-of-n concept can be represented by the following discrimination function:

$$f^{m,n}(\mathbf{x}) = \sum_{i=1}^{n} x_i - m.$$

Domingos and Pazzani [1997] test the naive Bayes classifier for m-of-n concepts learned from the complete, noise-free dataset and observed that the maximum-likelihood estimation of the parameters yields, in some cases, to non-optimal classifiers under 0-1 loss.

In general consider the *true* probability to be P such that $\Delta^2_{\mathcal{G}} f_{\mathrm{P}} \equiv 0$ for a given graph $\mathcal{G}$. We know from Theorem 5.1 that there exist a $\mathcal{G}$-classifier $\hat{\mathrm{P}}$ such that $f_{\hat{\mathrm{P}}} = f_{\mathrm{P}}$, but the maximum-likelihood estimation (even asymptotically) is not able to learn it and more importantly is not even able to learn a sign-equivalent function.

We tested this statement numerically in the simplest case of the naive Bayes model over two binary predictors. In particular, for a given value $\alpha > 0$ we randomly generate two conditional probabilities $\mathrm{P}(X_1, X_2|C = \pm 1) \in \mathcal{M}(\alpha)$ and we thus define $\mathrm{P}(X_1, X_2, C) = \frac{1}{2}\mathrm{P}(X_1, X_2|C)$. We then compute $\hat{\mathrm{P}} = \mathrm{P}(C)\,\mathrm{P}(X_1|C)\,\mathrm{P}(X_2|C)$ the oracle naive Bayes estimation and we test if $sign(f_{\hat{\mathrm{P}}}) = sign(f_{\mathrm{P}})$. For $\alpha = 1$, the true probability satisfies indeed the naive Bayes assumption and thus $sign(f_{\hat{\mathrm{P}}})$ makes no errors, but as α get increasingly far from one the NB estimation incurs more frequently in errors.

Modified Naive Bayes

Domingos and Pazzani [1997] reported that a simple modification of the naive Bayes allows to perfectly represent all of m-of-n concepts (from empirical evaluation for $n < 18$). The modification consists of adding a constant to the learned discrimination function $f_{\mathrm{P}^{nb}}$. The value of the constant is chosen to maximize the accuracy over the training data. We call such learning algorithm wNB (it is equivalent to the so-called weighted naive Bayes, see Chapter 3 for the explanation of such equivalence).

Using the odds-ratio parametrization we performed extensively evaluation of wNB for the case of two binary predictor variables in the oracle setting as specified above. As stated before, we observe that the NB model incurs in more errors when the odds ratio is far from 1, while the wNB corrected algorithm is always able to perfectly learn f_{P}.

Observe that the setting of Domingos and Pazzani [1997] is different from ours, their *true* probability is not a member of $\mathcal{P}$ since it is zeros if $C \neq f^{m,n}(\mathbf{X})$, it is an example of what we called a deterministic classification problem in Example 5.1.

We also tested extensively the wNB model with deterministic datasets obtained from linear discrimination function of the form $f = \sum_{i=1}^{n} \beta_i x_i + \gamma$ and with $\mathrm{P}(\mathbf{X}) =$

$\prod_{i=1}^{n} \mathrm{P}(X_i)$, $|\mathcal{X}_i| = m$, we observe that if $\beta_i = \beta$ for all $i \in [n]$ then the wNB model is always able to perfectly learn f. For general linear functions of $\mathbf{X}_i$ this is not the case thus showing that wNB is still limited in the type of discrimination functions learnable.

The weighted strategy has been employed successfully in the literature, both with naive Bayes models and with more complex structures. Recently Zaidi et al. [2017] studied a general framework that adds exponential weights to the recursive BN factorization. They showed that such over parametrized models can be initialized with parameters learned with maximum-likelihood and then refined with discriminative learning, that is maximizing the conditional likelihood.

5.5.2 Fixed Discrimination Maximum-Likelihood Estimator

Fix an undirected graph $\mathcal{G}$ and a discrimination function $f \in \mathcal{F}$ such that $\Delta^2_{\mathcal{G}} f \equiv 0$. From Theorem 5.1, we have that there exists a classifier in $\mathcal{P}(\mathcal{G})$ that induces f (that is $\Psi(\mathrm{P}) = f_{\mathrm{P}} = f$). Actually, from the proof of Theorem 5.1, it follows that there exists a whole family of $\mathcal{G}$-Markov classifiers that induce f.

We are interested now in obtaining the generative classifier that maximizes the likelihood among such family. Similarly to the maximum-likelihood estimation in Markov models, we need to complete the set with the limiting distributions. Thus we look at the set $\overline{\mathcal{P}}_{\mathcal{G}}(f)$ (see Sec. 5.3.1), such that

$$\arg \max_{\mathrm{P} \in \overline{\mathcal{P}}_{\mathcal{G}}(f)} \mathcal{L}(\mathrm{P}; \mathcal{D}) = \arg \max_{\mathrm{P} \in \overline{\mathcal{P}}_{\mathcal{G}}(f)} \prod_{(\mathbf{x},c) \in \mathcal{D}} \mathrm{P}(\mathbf{X} = \mathbf{x}, C = c)$$

We show how the iterative proportional fitting (IPF) algorithm [Fienberg, 1970, Lauritzen, 1996] can be used to solve the problem.

Let $\mathcal{K}(\mathcal{G})$ the set of cliques of the graph $\mathcal{G}$, and $\mathrm{P} \in \mathcal{P}_{\mathcal{G}} \cap \Psi^{-1}(f)$. For $A \in \mathcal{K}(\mathcal{G})$ we define the marginal fitting operator:

$$T_A \, \mathrm{P}(\mathbf{X} = \mathbf{x}, C = c) = \mathrm{P}(\mathbf{X} = \mathbf{x}, C = c) \frac{N_{\mathcal{D}}(\mathbf{x}_A)/|\mathcal{D}|}{\mathrm{P}(\mathbf{X}_A = \mathbf{x}_A)}$$

Observe that,

$$f_{T_A \mathrm{P}} = \log \left(\frac{T_A \, \mathrm{P}(\mathbf{X} = \mathbf{x}, C = +1)}{T_A \, \mathrm{P}(\mathbf{X} = \mathbf{x}, C = -1)} \right) = \log \left(\frac{\mathrm{P}(\mathbf{X} = \mathbf{x}, C = +1)}{\mathrm{P}(\mathbf{X} = \mathbf{x}, C = -1)} \right) = f_{\mathrm{P}}.$$

Thus $T_A \, \mathrm{P} \in \mathcal{P}_{\mathcal{G}} \cap \Psi^{-1}(f)$.

Given an ordering of the cliques $\mathcal{KG}$ the IPF algorithm iteratively adjusts the marginal of the cliques until convergence (see Section 2.2.1).

We just have to assure to initialize the IPF algorithm with a probability in $\mathcal{P}(\mathcal{G}) \cap \Psi^{-1}(f)$, for example,

$$\mathrm{P}^0(\mathbf{X} = \mathbf{x}, C = c) \propto \exp\left(\frac{c}{2} f(\mathbf{x})\right).$$

It is obvious that the resulting maximum-likelihood estimation will yield an element of $\overline{\mathcal{P}}_{\mathcal{G}}(f)$.

Combining Discriminative and Generative Approaches

It is obvious now how to combine the discriminative and generative approaches. Suppose we have a dataset $\mathcal{D}$ and a discriminative algorithms, that is an estimated function $\hat{f}_{\mathcal{D}} \in \mathcal{F}$. Suppose that for a given graph $\mathcal{G}$ we know that $\Delta^2_{\mathcal{G}} \hat{f}_{\mathcal{D}} \equiv 0$. We can find, using the IPF algorithm, the maximum likelihood $\mathcal{G}$-Markov classifier that induces $\hat{f}_{\mathcal{D}}$.

Observe that for various classical discriminative algorithms it is possible to know a priori the graph $\mathcal{G}$ such that $\Delta^2_{\mathcal{G}} \hat{f}_{\mathcal{D}} \equiv 0$. That is because a decomposition of $\hat{f}_{\mathcal{D}}$ is known as

$$\hat{f}_{\mathcal{D}}(\mathbf{x}) = \sum_{A} g_A(\mathbf{x}_A).$$

Examples include logistic regression and in general log-linear models, support vector machines [Cortes and Vapnik, 1995] and boolean classifiers (e.g using monomials basis) [O'Donnell, 2014].

5.6 Conclusions

In this chapter we analyze the impact of conditional independence statements and in general of the undirected Markov property over the induced discrimination function of a generative classifier. For this, we define a categorical differential operator (Δ_A) and we show that conditional independence statements are described by second-order differences being zero ($\Delta_A \Delta_B f \equiv 0$) for the induced discrimination function. We then connect such second order differences with multi-dimensional odds ratios and we study some ideas that such a formalization can suggest for learning algorithms. We think that the given descriptions of conditional independence statements for discrimination functions could be useful to study generative classifiers over categorical predictors and to help design new type of learning procedures.

Chapter 6

Conclusions

6.1 Summary of Contributions

The contributions of this thesis have been described in Chapters 3, 4 and 5. Chapter 3 includes our main results about the description of the discrimination functions induced by BAN classifiers. In particular for every BAN structure we are able to find a family of polynomials that interpolate the induced discrimination functions and thus sign-represent the associated decision functions. When the BAN structures do not contain V-structures we are also able to prove that every polynomial in that family can be induced by a BAN classifier with a given structure. Thanks to this polynomial description we are able to bound the number of decision functions representable by a given BAN structure, thus extending previous results in the literature.

Chapter 4 extends the study of decision functions induced by BAN classifiers to the multi-label case. In particular we study two simple but common methods, namely binary relevance and chain classifiers when BAN classifiers are used as base models. We are able to extend the bounding of the number of decision functions representable for these two multi-label methods. Moreover we prove that the chain method greatly expands the number of decision functions representable with respect to the binary relevance method.

In Chapter 5 we study generative classifiers under conditional independence assumptions and in general undirected Markov property. We connect undirected Markov properties with a set of linear relationships for the induced discrimination functions. Such linear relationships are formally described using a categorical difference operator. We show how this novel formalization for discrimination functions is useful to better understand generative classifiers and their induced decisions. Moreover we obtain some ideas for an alternative estimation of generative models and for combining discriminative and generative approaches.

6.2 List of Publications

The publications and submissions derived from the research reported in the present dissertation are listed below.

JCR articles

- G. Varando, C. Bielza, P. Larrañaga. Decision boundary for Bayesian network classifiers. *Journal of Machine Learning Research*, vol. 16, pp. 2725-2749, 2015.

- H. Borchani, G. Varando, C. Bielza, and P. Larrañaga, A survey on multi-output regression. *WIREs Data Mining and Knowledge Discovery*, vol. 5, pp. 216–233, 2015.

- G. Varando, C. Bielza, P. Larrañaga. Decision functions for chain classifiers based on Bayesian networks for multi-label classification. *International Journal of Approximate Reasoning*, vol. 68, pp. 164-178, 2016.

Submission

- G. Varando, E. Riccomagno, C. Bielza, P. Larrañaga. Markov property in generative classifiers, *to be submitted* 2018.

In proceedings

- G. Varando, C. Bielza, P. Larrañaga. Expressive power of binary relevance and chain classifiers based on Bayesian networks for multi-label classification. Lecture Notes in Artificial Intelligence, 8754, Springer, pp. 519-534, 2014.

Chapter 3 is directly derived with few changes from Varando et al. [2015]. Chapter 4 includes mainly the content of Varando et al. [2016] and Varando et al. [2014] plus an additional section containing a result published in Borchani et al. [2015]. Chapter 5 contains the work that is currently under preparation as Varando et al. 2018.

6.3 Future Work

In the conclusion sections at the end of every chapter we have already suggested some possible lines for future research. We now state more general questions and future research lines.

Learning algorithms We think that the results in the present dissertation give a solid framework for the study of generative classifiers over categorical predictors. We would now use the developed tools and the learned knowledge to design new types of learning algorithms for generative classifiers. In Chapter 5 we suggested a couple of possible approaches, but further analysis and empirical experimentation are needed. In particular we would like to investigate two different aspects of the learning procedure, namely fitting of parameter and structure search (or model selection). As far as parameter fitting is concerned, we want to further examine and implement methods that combine generative and discriminative approaches and test them over examples with missing data.

As far as model selection is concerned, it would be interesting to further study how to select the *best* BN structure (or undirected graph in Markov models) to address a classification problem. In particular it would be interesting to implement structural risk minimization model selection, especially in combination with a discriminative-generative approach, for classifiers based on graphical models.

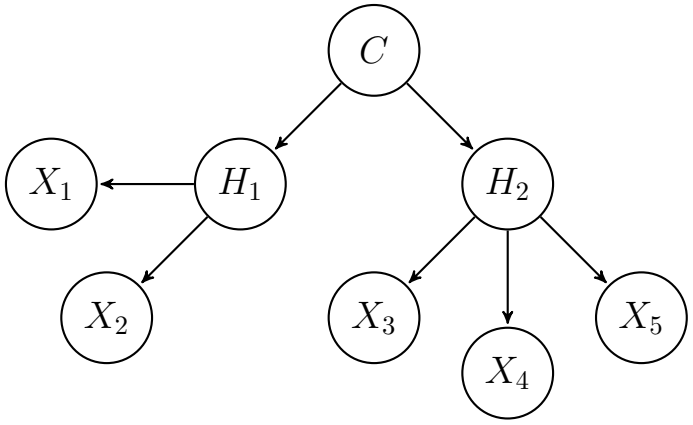


Figure 6.1: A hierarchical naive Bayes structure with five predictors and two hidden variables.

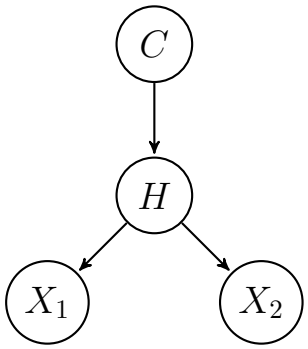


Figure 6.2: Simplest hierarchical naive Bayes over two predictors.

Hidden variables Hidden variables, that is, variables that are not observed not even in the training set, could be a valuable tool to expand the capabilities of BN classifiers and graphical model classifiers in general. In the literature the most studied model is the so-called *hierarchical naive Bayes* classifier [Han et al., 2005, Langseth and Nielsen, 2006, Flores et al., 2009, Njah et al., 2016], where the hidden variables are usually placed between the class variable and the predictors (see Figure 6.1). Langseth and Nielsen [2006] showed an example of a decision function over three binary predictors which is not representable by a naive Bayes but it is when a latent variable is added. It is easier to see, in general, that if we consider a single hidden variable between the predictors and the class as in Figure 6.2 and we do not place a bound on the number of values variable H can assume, it is obvious that the model can always represent whatsoever classifier (just by representing with the hidden variable H the product space of the predictors).

The first step would be to extend the results of Theorem 3.1 and in general Theorems 3.3 and 5.1 to generative models with hidden variables.

From an intuitive point of view, placing hidden variables is somehow the equivalent of the hidden layer in artificial neural networks (observe that the naive Bayes model is equivalent to the simple perceptron[Rosenblatt, 1957]). Also for this reason exploring the meaning of hidden variables for BN classifiers would be extremely interesting.

Bibliography

Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables.* Applied Mathematics Series. Dover Publications, 1964.

Concha Bielza and Pedro Larrañaga. Discrete Bayesian network classifiers: A survey. *ACM Comput. Surv.*, 47(1):5:1–5:43, 2014.

Concha Bielza, Guangdi Li, and Pedro Larrañaga. Multi-dimensional classification with Bayesian networks. *International Journal of Approximate Reasoning*, 52:705–727, 2011.

Hendrik Blockeel, Leander Schietgat, Jan Struyf, Sašo Džeroski, and Amanda Clare. Decision trees for hierarchical multilabel classification: A case study in functional genomics. In Johannes Fürnkranz, Tobias Scheffer, and Myra Spiliopoulou, editors, *Knowledge Discovery in Databases: PKDD 2006*, volume 4213 of *Lecture Notes in Computer Science*, pages 18–29. Springer Berlin Heidelberg, 2006.

Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larrañaga. A survey on multi-output regression. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(5):216–233, 2015. ISSN 1942-4795.

Enrico Carlini and Fabio Rapallo. The geometry of statistical models for two-way contingency tables with fixed odds ratios. *Rendiconti dell'Instituto di Matematica dell'Università di Trieste*, 37:71–84, 2005.

Jie Cheng and Russell Greiner. Comparing Bayesian network classifiers. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI'99, pages 101–108. Morgan Kaufmann Publishers Inc., 1999.

Jie Cheng and Russell Greiner. Learning Bayesian belief network classifiers: Algorithms and system. In *Proceedings of the 14th Biennial Conference of the Canadian Society on Computational Studies of Intelligence: Advances in Artificial Intelligence*, AI '01, pages 141–151. Springer-Verlag, 2001.

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

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20 (3):273–297, 1995.

Peter R. de Waal and Linda C. van der Gaag. Inference and learning in multi-dimensional Bayesian network classifiers. In Khaled Mellouli, editor, *ECSQARU*, volume 4724 of *Lecture Notes in Computer Science*, pages 501–511. Springer, 2007.

Krzysztof Dembczynski, Weiwei Cheng, and Eyke Hüllermeier. Bayes optimal multilabel classification via probabilistic classifier chains. In Johannes Fürnkranz and Thorsten Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 279–286. Omnipress, 2010.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, 1996.

Persi Diaconis and Bernd Sturmfels. Algebraic algorithms for sampling from conditional distributions. *Annals of Statistics*, 26:363–397, 1995.

Pedro Domingos and Michael Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130, 1997.

M. Drton, B. Sturmfels, and S. Sullivant. *Lectures on Algebraic Statistics*. Oberwolfach Seminars. Birkhäuser Basel, 2009.

Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

Stephen E. Fienberg. The geometry of an $r \times c$ contingency table. *The Annals of Mathematical Statistics*, 39(4):1186–1190, 1968.

Stephen E. Fienberg. An iterative procedure for estimation in contingency tables. *Ann. Math. Statist.*, 41(3):907–917, 1970.

Stephen E. Fienberg and John P. Gilbert. The geometry of a two by two contingency table. *Journal of the American Statistical Association*, 65(330):694–701, 1970.

Leopold Flatto. A new proof of the transposition theorem. *Proceedings of the American Mathematical Society*, 24(1):29–31, 1970.

M. Julia Flores, José A. Gámez, Ana M. Martínez, and José M. Puerta. Hode: Hidden one-dependence estimator. In Claudio Sossai and Gaetano Chemello, editors, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 481–492. Springer Berlin Heidelberg, 2009.

Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.

Alberto Gandolfi and Pietro Lenarda. A note on gibbs and markov random fields with constraints and their moments. *Mathematics and Mechanics of Complex Systems*, 4 (3-4):407–422, 2017.

Luis David Garcia, Michael Stillman, and Bernd Sturmfels. Algebraic geometry of Bayesian networks. *Journal of Symbolic Computation*, 39(3):331–355, 2005. Special issue on the occasion of MEGA 2003.

Teresa Gonçalves and Paulo Quaresma. A preliminary approach to the multilabel classification problem of Portuguese juridical documents. In FernandoMoura Pires and Salvador Abreu, editors, *Progress in Artificial Intelligence*, volume 2902 of *Lecture Notes in Computer Science*, pages 435–444. Springer Berlin Heidelberg, 2003.

Mark Hall. A decision tree-based attribute weighting filter for naive Bayes. In Max Bramer, Frans Coenen, and Andrew Tuson, editors, *Research and Development in Intelligent Systems XXIII*, pages 59–70. Springer London, 2007.

John Hammersley and Peter Clifford. Markov fields on finite graphs and lattices. 1971. Unpublished manuscript.

Hui Han, Wei Xu, Hongyuan Zha, and C. Lee Giles. A hierarchical naive Bayes mixture model for name disambiguation in author citations. In *SAC*, 2005.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Kenneth E. Iverson. *A Programming Language*. John Wiley & Sons, Inc., 1962.

Manfred Jaeger. Probabilistic classifiers and the concepts they recognize. In Tom Fawcett and Nina Mishra, editors, *Proceedings of the Twentieth International Conference on Machine Learning (ICML-03)*, pages 266–273. AAAI Press, 2003.

Harold Jeffreys and Bertha Jeffreys. *Methods of Mathematical Physics*. Cambridge Mathematical Library. Cambridge University Press, 1999.

Marcel Jung, Octavian Niculita, and Zakwan Skaf. Comparison of different classification algorithms for fault detection and fault isolation in complex systems. *Procedia Manufacturing*, 19:111 – 118, 2018. Proceedings of the 6th International Conference in Through-life Engineering Services, University of Bremen, 7th and 8th November 2017.

Eamonn J. Keogh and Michael J. Pazzani. Learning the structure of augmented Bayesian classifiers. *International Journal on Artificial Intelligence Tools*, 11(04):587–601, 2002.

Helge Langseth and Thomas D. Nielsen. Classification using hierarchical naïve Bayes models. *Machine Learning*, 63(2):135–159, 2006.

Julia Lasserre, Christopher M. Bishop, J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West. *Generative or Discriminative? Getting the Best of Both Worlds*, volume 8, pages 3–24. Oxford University Press, 2007.

Steffen L. Lauritzen. *Graphical Models*. Oxford Statistical Science Series. Clarendon Press, 1996.

Charles X. Ling and Huajie Zhang. The representational power of discrete Bayesian networks. *Journal of Machine Learning Research*, 3:709–721, 2002.

Marvin Minsky. Steps toward artificial intelligence. In *Computers and Thought*, pages 406–450. McGraw-Hill, 1961.

Dinora A. Morales, Yolanda Vives-Gilabert, Beatriz Gómez-Ansón, Endika Bengoetxea, Pedro Larrañaga, Concha Bielza, Javier Pagonabarraga, Jaime Kulisevsky, Idoia Corcuera-Solano, and Manuel Delfino. Predicting dementia development in Parkinson's disease using Bayesian network classifiers. *Psychiatry Research: Neuroimaging*, 213(2):92 – 98, 2013.

Atsuyoshi Nakamura, Michael Schmitt, Niels Schmitt, and Hans Ulrich Simon. Inner product spaces for Bayesian networks. *Journal of Machine Learning Research*, 6: 1383–1403, 2005.

Andrew Y. Ng and Michael I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, pages 841–848. MIT Press, 2001.

H. Njah, S. Jamoussi, and W. Mahdi. Semi-hierarchical naive Bayes classifier. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 1772–1779, 2016.

Ryan O'Donnell. *Spectral Structure and Learning*, page 54–78. Cambridge University Press, 2014.

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

Mark A. Peot. Geometric implications of the naive Bayes assumption. In *Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence*, UAI'96, pages 414–419. Morgan Kaufmann Publishers Inc., 1996.

Giovanni Pistone, Eva Riccomagno, and Henry P. Wynn. Gröbner bases and factorisation in discrete probability and Bayes. *Statistics and Computing*, 11(1):37–46, 2001.

Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. Classifier chains for multi-label classification. In *Machine Learning and Knowledge Discovery in Databases*, volume 5782 of *Lecture Notes in Computer Science*, pages 254–269. Springer Berlin Heidelberg, 2009.

Jesse Read, Concha Bielza, and Pedro Larrañaga. Multi-dimensional classification with super-classes. *IEEE Transactions on Knowledge and Data Engineering*, 26(7):1720–1733, 2014. ISSN 1041-4347.

Jesse Read, Luca Martino, Pablo M. Olmos, and David Luengo. Scalable multi-output label prediction: From classifier chains to classifier trellises. *Pattern Recognition*, 48 (6):2096 – 2109, 2015.

Teemu Roos, Hannes Wettig, Peter Grünwald, Petri Myllymäki, and Henry Tirri. On discriminative Bayesian network classifiers and logistic regression. *Machine Learning*, 59(3):267–296, 2005.

Frank Rosenblatt. *The Perceptron, a perceiving and recognizing automaton*. Cornell Aeronautical Laboratory, 1957.

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

Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A Bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the 1998 Workshop*, volume 62, pages 98–105, 1998.

Raffaella Settimi and Jim Q. Smith. On the geometry of Bayesian graphical models with hidden variables. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, pages 472–479. Morgan Kaufmann Publishers Inc., 1998.

L. Enrique Sucar, Concha Bielza, Eduardo F. Morales, Pablo Hernandez-Leal, Julio H. Zaragoza, and Pedro Larrañaga. Multi-label classification with Bayesian network-based chain classifiers. *Pattern Recognition Letter*, 41:14–22, 2014.

Grigorios Tsoumakas and Ioannis Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 2007:1–13, 2007.

Linda C. van der Gaag and Peter R. de Waal. Multi-dimensional Bayesian network classifiers. In Milan Studený and Jirí Vomlel, editors, *Third European Workshop on Probabilistic Graphical Models*, pages 107–114, 2006.

Vladimir N. Vapnik and Alexy Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16(2):264–280, 1971.

Gherardo Varando, Concha Bielza, and Pedro Larrañaga. Expressive power of binary relevance and chain classifiers based on Bayesian networks for multi-label classification. In Linda C. van der Gaag and Ad J. Feelders, editors, *Probabilistic Graphical Models*, volume 8754 of *Lecture Notes in Computer Science*, pages 519–534. Springer, 2014.

Gherardo Varando, Concha Bielza, and Pedro Larranaga. Decision boundary for discrete Bayesian network classifiers. *Journal of Machine Learning Research*, 16:2725–2749, 2015.

Gherardo Varando, Concha Bielza, and Pedro Larrañaga. Decision functions for chain classifiers based on Bayesian networks for multi-label classification. *Int. J. Approx. Reasoning*, 68:164–178, 2016.

Linda Varghese, V. I. George, Krishnamoorthi Makkithaya, and Abhishek Kumar. Data driven approach to monitoring and fault detection in process control plants. *International Journal of Control Theory and Applications*, 8(3):1121–1128, 2015.

Geoffrey I. Webb and Michael J. Pazzani. Adjusted probability naive Bayesian induction. In *Proceedings of the Eleventh Australian Joint Conference on Artificial Intelligence*, pages 285–295. Springer-Verlag, 1998.

Youlong Yang and Yan Wu. On the properties of concept classes induced by multivalued Bayesian networks. *Information Sciences*, 184(1):155–165, 2012.

Nayyar A. Zaidi, Jesus Cerquides, Mark J. Carman, and Geoffrey I. Webb. Alleviating naive Bayes attribute independence assumption by attribute weighting. *Journal of Machine Learning Research*, 14:1947–1988, 2013.

Nayyar A. Zaidi, Geoffrey I. Webb, Mark J. Carman, François Petitjean, Wray Buntine, Mike Hynes, and Hans De Sterck. Efficient parameter learning of Bayesian network classifiers. *Machine Learning*, 106(9):1289–1329, 2017.

Min-Ling Zhang and Zhi-Hua Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038 – 2048, 2007.

Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.