Model selection for discrete Markov random fields on graphs

Iara Moreira Frondana

PhD thesis presented
to
Institute of Mathematics and Statistics
of
University of São Paulo
to
obtain the title
of
Doctor in Science

Program: Statistics

Advisor: Prof. Florencia Graciela Leonardi

During the development of this work the author received financial support from CAPES and CNPq. This thesis was produced as part of the activities of the Research, Innovation and Dissemination Center for Neuromathematics, funded by FAPESP (grant 2013/07699-0).

São Paulo, May 2016

Seleção de modelos para campos aleatórios Markovianos discretos sobre grafos

Iara Moreira Frondana

Tese apresentada
ao
Instituto de Matemática e Estatística
da
Universidade de São Paulo
para
obtenção do título
de
Doutora em Ciências

Programa: Estatística

Orientadora: Profa. Dra. Florencia Graciela Leonardi

Durante o desenvolvimento deste trabalho a autora recebeu auxílio financeiro da CAPES e CNPq. Esta tese foi produzida como parte das atividades do Centro de Pesquisa, Inovação e Difusão em Neuromatemática, financiado pela FAPESP (processo 2013/07699-0).

São Paulo, maio de 2016

Model selection for discrete Markov random fields on graphs

This is the original version of the thesis elaborated by
candidate Iara Moreira Frondana, such as
submitted to the Judging Committee.

Seleção de modelos para campos aleatórios Markovianos discretos sobre grafos

Esta é a versão original da tese elaborada pela candidata Iara Moreira Frondana, tal como submetida à Comissão Julgadora.

Abstract

FRONDANA, I. M. **Model selection for discrete Markov random fields on graphs**. PhD Thesis - Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2016.

In this thesis we propose to use a penalized maximum conditional likelihood criterion to estimate the graph of a general discrete Markov random field. We prove the almost sure convergence of the estimator of the graph in the case of a finite or countable infinite set of variables. Our method requires minimal assumptions on the probability distribution and contrary to other approaches in the literature, the usual positivity condition is not needed. We present several examples with a finite set of vertices and study the performance of the estimator on simulated data from theses examples. We also introduce an empirical procedure based on $k-$fold cross validation to select the best value of the constant in the estimator's definition and show the application of this method in two real datasets.

Keywords: simple undirected graphs, discrete Markov random fields, model selection, Bayesian information criterion.

Resumo

FRONDANA, I. M. **Seleção de modelos para campos aleatórios Markovianos discretos sobre grafos**. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2016.

Nesta tese propomos um critério de máxima verossimilhança penalizada para estimar o grafo de dependência condicional de um campo aleatório Markoviano discreto. Provamos a convergência quase certa do estimador do grafo no caso de um conjunto finito ou infinito enumerável de variáveis. Nosso método requer condições mínimas na distribuição de probabilidade e contrariamente a outras abordagens da literatura, a condição usual de positividade não é necessária. Introduzimos alguns exemplos com um conjunto finito de vértices e estudamos o desempenho do estimado em dados simulados desses exemplos. Também propomos um procedimento empírico baseado no método de validação cruzada para selecionar o melhor valor da constante na definição do estimador, e mostramos a aplicação deste procedimento em dois conjuntos de dados reais.

Palavras-chave: grafos simples não-dirigidos, campos aleatórios Markovianos discretos, seleção de modelos, critério da Informação Bayesiano.

Contents

List of Figures

List of Tables

Chapter 1

Indroduction

In this thesis we study the problem of model selection for discrete Markov random fields on graphs. These models, also known as graphical models (Lauritzen, 1996) and probabilistic graphical models (Koller and Friedman, 2009), have received much attention from researchers in recent years, especially due to its flexibility to capture conditional dependence between variables. They have been applied to many different problems in different fields such as Biology (Shojaie and Michailidis, 2010) or Social Sciences (Strauss and Ikeda, 1990).

Classically, Markov random fields are defined over a lattice (as for example $\mathbb{Z}^d$, $d \geq 1$) and the conditional probability distributions of a variable given the remaining variables are assumed to be translation invariant. The Markov property establishes that these conditional probabilities depend only on a finite set of nearest variables (called Markov neighborhood). In our case, we relax this condition and allow the conditional probability distributions to depend on the specific variable. For this reason the Markov neighborhoods are no longer translation invariant but define a simple and undirected graph. In this thesis we focus the analysis on discrete models, that is the set of random variables takes values on a finite alphabet.

One of the main statistical problems for this type of models is to estimate the underlying graph; that is, the graph determined by the conditional dependence relationships between the variables. For the class of Markov random fields on lattices, some methods based on penalized maximum likelihood criteria like the Bayesian Information Criterion (BIC) of Schwarz (1978) have appeared in the literature (Csiszár and Talata, 2006). For the general class of models defined on graphs, there are some works addressing the problem of model selection but based on other methods. An example of this are the estimators based on the *Least Absolute Shrinkage and Selector Operator* - LASSO, (Tibshirani, 1996). The LASSO has been applied to estimate the graph for normally distributed variables (Meinshausen and Bühlmann (2006) or for discrete binary alphabets (Ravikumar et al., 2010). Other methods are based on pairwise dependence determination between variables for discrete binary alphabets (Galves et al., 2015) or based on properties of the inverse covariance matrix (Loh and Wainwright, 2013), for general discrete variables.

In general, these works address the problem of consistency of the proposed estimators. In Csiszár and Talata (2006), the consistency of the penalized maximum pseudo-likelihood criterion is proved based on a unique realization of the process. This is only possible due to the translation invariant characteristic of the transition probabilities. In the other works and for general models defined on graphs, the model selection is based on independent realizations of the joint distribution. In the case of the methods based on the LASSO, the proof of the consistency is related to the identification of the non-zero coefficients of a linear regression or logistic (in the binary case) model. One drawback of this approach is the amount of regularity assumptions about the probability distributions, which are practically impossible to be verified in practice. In the case of the approach presented in Galves et al. (2015), the consistency criterion is specific for binary variables and can not be generalized to the general discrete case. The last method is intended for discrete graphical models belonging to the exponential family but is developed only for the special case of multinomial distributions.

In this thesis we propose to use a penalized maximum conditional likelihood criterion to estimate the graph of a general discrete Markov random field. We prove the almost sure convergence of the estimator of the graph in the case of a finite or countable infinite set of variables. Our method requires minimal assumptions on the probability distribution and contrary to other approaches in the literature, the usual positivity condition is not needed. We present several examples with a finite set of vertices and study the performance of the estimator on simulated data from theses examples. We also introduce an empirical procedure based on $k-$fold cross validation to select the best value of the constant in the estimator's definition and show the application of this method in two real datasets.

This thesis is structured as follows: Chapter 2 presents the definition of the model and the estimator in the case of a finite set of vertices. It also includes some examples and the proof of the consistency of the estimator in this particular case. In Chapter 3 we evaluate the estimator performance through simulations, considering different aspects, such as the sample size, the value of the constant appearing in the definition of the estimator and the choice between a conservative or non conservative approach for the estimation of the full graph. Taking into account the results obtained in the previous chapter, Chapter 4 introduces the empirical method to select the best value of the constant in the estimator's definition and shows the application to the real datasets. In Chapter 5 we revisit the theoretical problem and prove the consistency of a generalized estimator when the set of vertices is countable infinite. Our conclusions are presented in Chapter 6, and Appendices A and B contain some theoretical results needed in this thesis and the R programs used, respectively.

Chapter 2

Discrete Markov random fields on graphs

This chapter presents the definition of a discrete Markov random field on A^V with a graph G along with all the necessary definitions for the correct understanding of the subject. They are also presented some basic lemmas, as well as some examples. Then we show how to estimate the probabilities in a Markov random fiel on A^V with graph G when the graph structure is know, and how to estimate the graph through the estimation of the vertices' neighbors using a penalized maximum conditional likelihood method similar to the Bayesian Information Criterion (BIC) of Schwarz (1978).

2.1 Definitions

A *graph* is a pair $G = (V, E)$, where V is the set of vertices and E is the set of edges, $E \subset V \times V$. A graph G is said *simple* if for all $i \in E$, $(i, i) \notin E$ and it is said *undirected* if $(i, j) \in E$ implies $(j, i) \in E$ for all pair $(i, j) \in V \times V$. From this moment on, the word graph is used to denote a simple undirected graph. Given any set S, the symbol $|S|$ denotes the cardinality of S, where S is a set of vertices or edges.

Let A be a finite set, a *random field* over A^V is a family of random variables indexed by the elements of V, $\{X_v : v \in V\}$, where each X_v is a variable with values in A. For $\Delta \subseteq V$, a subset of vertices, we write $X_\Delta = \{X_i \colon i \in \Delta\}$, and $a_\Delta = \{a_i \in A \colon i \in \Delta\}$ denotes a configuration on Δ. The joint distribution of the random variables X_i (which it is assumed to exist) is denoted by $\mathbb{P}$:

$$\mathbb{P}(a_\Delta) = \mathbb{P}(X_\Delta = a_\Delta) \text{ for } \Delta \subset V \text{ finite}, a_\Delta \in A^\Delta.$$

And the definition of the conditional probabilities is

$$\mathbb{P}(a_\Delta | a_\Phi) = \mathbb{P}(X_\Delta = a_\Delta | X_\Phi = a_\Phi) \text{ for } a_\Delta \in A^\Delta, a_\Phi \in A^\Phi, \mathbb{P}(a_\Phi) > 0$$

for all finite disjoint subsets $\Delta, \Phi \subset V$. From now on, when we write $\mathbb{P}(a_\Delta | a_\Phi)$ we assume that $\mathbb{P}(a_\Phi) > 0$.

A *neighborhood* W of v is any finite set of vertices with $v \notin W$. And a discrete *Markov random field* is a random field as described above such that, for all $v \in V$, there is a neighborhood W of v, called *Markov neighborhood*, satisfying

$$\mathbb{P}(a_v | a_\Delta) = \mathbb{P}(a_v | a_W) \tag{2.1}$$

for all $\Delta \supset W$ finite, $v \notin \Delta$ and all $a_v \in A$, $a_\Delta \in A^\Delta$ with $\mathbb{P}(a_\Delta) > 0$.

The definition of the Markov neighborhood W is equivalent to request that for all Φ finite with $\Phi \cap W = \emptyset$, X_Φ is conditionally independent of X_v, given X_W. More formally,

$$X_v \perp\!\!\!\perp X_\Phi | X_W \,, \text{ for all } \Phi \text{ with } \Phi \cap W = \emptyset, \tag{2.2}$$

where $\perp\!\!\!\perp$ is the well known symbol of independence. This corresponds to the property known as *local Markov*, and it is weaker than the usually assumed *global Markov* property, see Lauritzen (1996) for details.

A basic fact that we can derive from the definition is that if W is a Markov neighborhood of $v \in V$ then any finite set $\Delta \supset W$ is also a Markov neighborhood of v. On the other hand, if W_1 and W_2 are Markov neighborhoods of v then it is not always true in general that $W_1 \cap W_2$ is a Markov neighborhood, as shown in the following example.

Example 1. *Let $V = \{1, 2, 3\}$ and consider the vector (X_1, X_2, X_3) of Bernoulli random variables with $\mathbb{P}(X_i = 0) = 1/2$, $\mathbb{P}(X_i = 1) = 1/2$, for $i = 1, 2, 3$. Suppose that $X_1 = X_2 = X_3$ with probability 1. Then it is easy to check that both $\{2\}$ and $\{3\}$ are Markov neighborhoods of node 1, but the intersection is not a Markov neighborhood (which will imply that X_1 is independent of X_2 and X_3).*

The intersection property for Markov neighborhoods is desirable in our context to define the smallest Markov neighborhood of a node and to enable the structure estimation problem to be well defined. For that reason we assume the distribution $\mathbb{P}$ satisfies the following:

Markov intersection property: For all $v \in V$ and all W_1 and W_2 Markov neighborhoods of v, the set $W_1 \cap W_2$ is also a Markov neighborhood of v.

This property is guaranteed under the usual positivity condition; namely that all marginal distributions of finite dimension are strictly positive, see Lauritzen (1996). But there are distributions satisfying the Markov intersection property that are not strictly positive. We borrow the following example from Lauritzen (1996), originally presented to show that in general the local and global Markov properties are not equivalent. This example also serves to show that positivity and Markov intersection are not equivalent in general.

Example 2. *Let $V = \{1, 2, 3, 4, 5\}$ and consider the random vector $(X_1, X_2, X_3, X_4, X_5)$ such that: X_1 and X_5 are independent, with $\mathbb{P}(X_1 = 0) = \mathbb{P}(X_1 = 1) = \mathbb{P}(X_5 = 0) = \mathbb{P}(X_5 = 1) = 1/2$, $X_2 = X_1$, $X_4 = X_5$ and $X_3 = X_2 X_4$. Then it can be seen that this distribution satisfies the Markov intersection property for all $v \in V$, but it does not satisfy the positivity condition.*

From now on and for the rest of this chapter we will focus on the case where V is a finite set. The case where V is countable infinite will be treated in Chapter 5. The next section presents the basic lemmas that define the smallest Markov neighborhood of a discrete Markov random field and prove that the graph $G = (V, E)$ cronstructed from these neighborhoods is undirected.

2.2 Basic lemmas

Let $\Theta(v)$ be the set of all subsets of V that are Markov neighborhoods of $v \in V$, i.e, $\Theta(v) = \{W_i \subset V : W_i \text{ is a Markov neighborhood of } v\}$. Note that because V is finite, the set $\Theta(v)$ is finite as well. Now, let define $\text{ne}(v)$, with $v \in V$, as

$$\text{ne}(v) = \bigcap_{W_i \in \Theta(v)} W_i. \tag{2.3}$$

Lemma 3. *For all $v \in V$ the subset $ne(v) \subset V$ defined by (2.3) is a Markov neighborhood of v.*

Proof. This lemma is a direct consequence of the Markov intersection property, noticing that $\text{ne}(v)$ is a finite intersection of Markov neighborhoods. □

By the definition of $\text{ne}(v)$ in (2.3) and Lemma 3, $\text{ne}(v)$ is the smallest Markov neighborhood of $v \in V$.

The next lemma is an adaptation of Lemma A.2 of Csiszár and Talata (2006).

Lemma 4. *For a Markov random field $\mathbb{P}$, if a neighborhood W satisfies*

$$\mathbb{P}(a_v|a_W) = \mathbb{P}(a_v|a_{ne(v)}) \text{ for all } a_V \in A^V$$

then W is a Markov neighborhood.

Proof. We have to show that

$$\mathbb{P}(a_v|a_{V\setminus\{v\}}) = \mathbb{P}(a_v|a_W) \text{ for all } a_V \in A^V. \tag{2.4}$$

As $\mathrm{ne}(v)$ is a Markov neighborhood, the lemma's condition implies that

$$\mathbb{P}(a_v|a_W) = \mathbb{P}(a_v|a_{\mathrm{ne}(v)}) = \mathbb{P}(a_v|a_{V\setminus\{v\}}) \text{ for all } a_V \in A^V.$$

So W is a Markov neighborhood. □

Given a discrete Markov random field over A^V, we define the graph $G = (V, E)$ by

$$(v, w) \in E \text{ if and only if } w \in \mathrm{ne}(v).$$

The graph G is usually known as *graph of interactions* of the discrete Markov random field.

Lemma 5. *The graph G, defined above, is undirected, i.e. if $(v, w) \in E \Rightarrow (w, v) \in E$.*

Proof. If $w \notin \mathrm{ne}(v)$ then

$$\mathbb{P}(a_v|a_{V\setminus\{v\}}) = \mathbb{P}(a_v|a_{V\setminus\{v,w\}}) \quad \text{for all } a_V \in A^V. \tag{2.5}$$

By the definition of conditional probability and (2.5) we have that

$$\begin{aligned} \mathbb{P}(a_v, a_w|a_{V\setminus\{v,w\}}) &= \mathbb{P}(a_v|a_w, a_{V\setminus\{v,w\}})P(a_w|a_{V\setminus\{v,w\}}) \\ &= \mathbb{P}(a_v|a_{V\setminus\{v,w\}})P(a_w|a_{V\setminus\{v,w\}}) \quad \text{for all } a_V \in A^V. \end{aligned} \tag{2.6}$$

But, on the other hand,

$$\mathbb{P}(a_v, a_w|a_{V\setminus\{v,w\}}) = \mathbb{P}(a_w|a_v, a_{V\setminus\{v,w\}})P(a_v|a_{V\setminus\{v,w\}}) \quad \text{for all } a_V \in A^V. \tag{2.7}$$

Therefore, by (2.6) and (2.7) we have

$$\mathbb{P}(a_w|a_v, a_{V\setminus\{v,w\}}) = P(a_w|a_{V\setminus\{v,w\}}) \text{ for all } a_V \in A^V \Longrightarrow v \notin \mathrm{ne}(w).$$

□

The conditional distribution of X_v given $X_{\mathrm{ne}(v)} = a_{\mathrm{ne}(v)}$ will be denoted by $\{p(a|a_{\mathrm{ne}(v)})\}_{a\in A}$; i.e., for all $a \in A$ we have

$$p(a|a_{\mathrm{ne}(v)}) = \mathbb{P}(X_v = a|X_k = a_k,\ k \in \mathrm{ne}(v))\,. \tag{2.8}$$

Similarly, the joint distribution of $(X_v, X_{\mathrm{ne}(v)})$ will be denoted by $\{p(a_v, a_{\mathrm{ne}(v)})\}_{a_v\in A, a_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}}$.

2.3 Examples

Here we present four examples of a Markov random field over A^V with graph G. These examples were constructed with different number of vertices, different conditional dependence structures and different alphabet sizes.

In each one of these examples the joint probabilities were defined as a product of conditional probabilities $p(a|a_W)$, where the set W does not have to be the neighborhood of the vertex. But the

choice of how the factorization of the joint probability is made defines the conditional dependency structures (i.e., defines the structure of the graph) and allows a simpler way of generating samples from Markov random fields over A^V with graph G (Chapter 3). For the first example we show how to find the neighborhood of each vertex through the chosen factorization, and this can be replicated to the examples 2, 3 and 4 as well.

Note that the theoretical values of the probabilities are also needed to generate the samples, so for each example these values were chosen without any particular reason, that is, other values could have been chosen.

2.3.1 Example 1

This first example has 5 random variables that assume values in $A = \{0, 1, 2\}$, and its joint probability is given by:

$$p(x_1, x_2, x_3, x_4, x_5) = p(x_2|x_1, x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3) \tag{2.9}$$

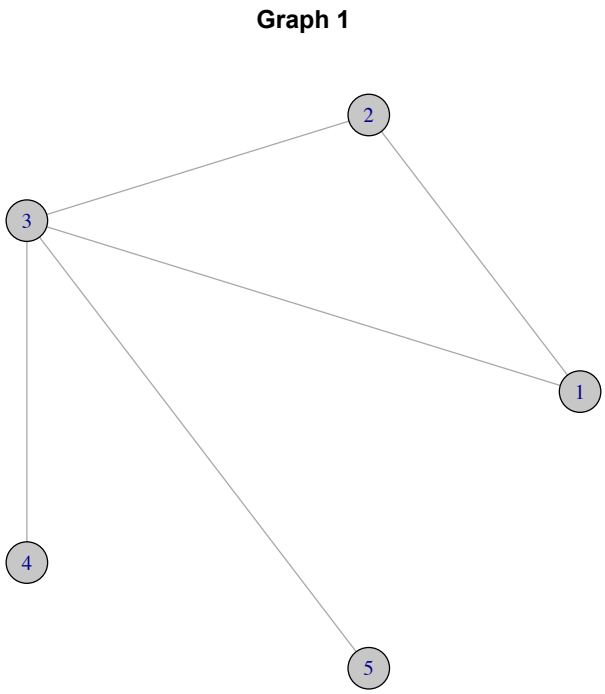

Figure 2.1: *Graph of example 1.*

To find the neighborhood of vertex 1 we have from 2.9:

$$\begin{aligned} p(x_1|x_2,x_3,x_4,x_5) &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)}{\sum_{x_1\in\{0,1,2\}} p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)} \\ &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)}{p(x_4|x_3)p(x_5|x_3)\sum_{x_1\in\{0,1,2\}} p(x_2|x_1,x_3)p(x_1|x_3)p(x_3)} \\ &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_3)}{\sum_{x_1\in\{0,1,2\}} p(x_2|x_1,x_3)p(x_1|x_3)p(x_3)} = \frac{p(x_2|x_1,x_3)p(x_1,x_3)}{\sum_{x_1\in\{0,1,2\}} p(x_2|x_1,x_3)p(x_1,x_3)} \\ &= \frac{p(x_1,x_2,x_3)}{\sum_{x_1\in\{0,1,2\}} p(x_1,x_2,x_3)} = \frac{p(x_1,x_2,x_3)}{p(x_2,x_3)} = p(x_1|x_2,x_3), \end{aligned} \tag{2.10}$$

so, by 2.10 the neighborhood of vertex 1 is $\{2,3\}$.

The neighborhood of vertex 2 is given by:

$$\begin{aligned} p(x_2|x_2,x_3,x_4,x_5) &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)}{\sum_{x_2\in\{0,1,2\}} p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)} \\ &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)}{p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)\sum_{x_2\in\{0,1,2\}} p(x_2|x_1,x_3)} \\ &= \frac{p(x_2|x_1,x_3)}{\sum_{x_2\in\{0,1,2\}} p(x_2|x_1,x_3)} = \frac{p(x_2|x_1,x_3)}{1} = p(x_2|x_1,x_3) \end{aligned} \tag{2.11}$$

then, by 2.11, the neighborhood of vertex 2 is $\{1,3\}$.

Now, the neighborhood of vertex 4 is given by:

$$\begin{aligned} p(x_4|x_2,x_3,x_4,x_5) &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)}{\sum_{x_4\in\{0,1,2\}} p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)} \\ &= \frac{p(x_2|x_1,x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)}{p(x_2|x_1,x_3)p(x_1|x_3)p(x_5|x_3)p(x_3)\sum_{x_4\in\{0,1,2\}} p(x_4|x_3)} \\ &= \frac{p(x_4|x_3)}{\sum_{x_4\in\{0,1,2\}} p(x_4|x_3)} = \frac{p(x_4|x_3)}{1} = p(x_4|x_3) \end{aligned} \tag{2.12}$$

then, by 2.12, the neighborhood of vertex 4 is $\{3\}$. Doing the same for vertex 5 we can find that its neighborhood is $\{3\}$. And, because vertex 3 is in the neighborhood of all other vertices, the neighborhood of vertex 3 is $\{1,2,4,5\}$. Figure 2.1 contains the graph that represents the neighborhoods of the vertices.

The theoretical values chosen for the probabilities in 2.9 are given below:

a) Marginal distribution of X_3:

x_3	0	1	2
$p(x_3)$	0.3	0.2	0.5

b) Conditional probabilities of $X_i|X_3 = x_3$, $i = 1,4,5$:

x_1	0	1	2
$p(x_1\|X_3=0)$	0.2	0.4	0.4
$p(x_1\|X_3=1)$	0.3	0.4	0.3
$p(x_1\|X_3=2)$	0.4	0.3	0.3

x_4	0	1	2
$p(x_4\|X_3=0)$	0.1	0.4	0.5
$p(x_4\|X_3=1)$	0.2	0.7	0.1
$p(x_4\|X_3=2)$	0.3	0.6	0.1

x_5	0	1	2
$p(x_5\|X_3=0)$	0.2	0.6	0.2
$p(x_5\|X_3=1)$	0.3	0.1	0.6
$p(x_5\|X_3=2)$	0.4	0.3	0.3

c) Conditional probability of $X_2|X_1=x_1, X_3=x_3$:

x_2	0	1	2
$p(x_2\|X_1=0,X_3=0)$	1/2	1/2	0
$p(x_2\|X_1=1,X_3=0)$	2/4	1/4	1/4
$p(x_2\|X_1=2,X_3=0)$	1/4	1/4	2/4
$p(x_2\|X_1=0,X_3=1)$	1/3	0	2/3
$p(x_2\|X_1=1,X_3=1)$	1/4	1/4	2/4
$p(x_2\|X_1=2,X_3=1)$	1/3	2/3	0
$p(x_2\|X_1=0,X_3=2)$	0	3/4	1/4
$p(x_2\|X_1=1,X_3=2)$	1/3	1/3	1/3
$p(x_2\|X_1=2,X_3=2)$	1/3	1/3	1/3

By the theoretical probabilities in (a-c) we can find the model's joint probability distribution for all values in A^5. For example, the probability of all variables equal to 0 is:

$$\begin{aligned} & p(X_1=0,X_2=0,X_3=0,X_4=0,X_5=0) \\ = \; & p(X_2=0|X_1=0,X_3=0)p(X_1=0|X_3=0)p(X_4=0|X_3=0)p(X_5=0|X_3=0)p(X_3=0) \\ = \; & 0.2\times 0.2\times 0.1\times 0.2\times 0.3 = 0.00024 \end{aligned}$$

Note that in item c) we have three probabilities equal to zero, this means that in this example we do not have all marginal distributions strictly positive.

2.3.2 Example 2

In this model we have 7 random variables that assume values in $A=\{0,1\}$, and its joint probability is given by:

$$p(x_1,x_2,x_3,x_4,x_5,x_6,x_7) = p(x_1|x_2)p(x_2|x_3)p(x_4|x_3)p(x_3)p(x_5|x_6)p(x_7|x_6)p(x_6). \quad (2.13)$$

Through 2.13 we can find the dependence structure of the model represented by the graph of Figure 2.2.

Below are presented the model's chosen theoretical probabilities:

a) Marginal distributions of X_3 and X_6:

x_3	0	1
$p(x_3)$	0.4	0.6

x_6	0	1
$p(x_6)$	0.6	0.4

b) Conditional probabilities of $X_i|X_3=x_3$, $i=2,4$ and $X_j|X_6=x_6$, $j=5,7$:

x_2	0	1
$p(x_2\|X_3=0)$	0.4	0.6
$p(x_2\|X_3=1)$	0.5	0.5

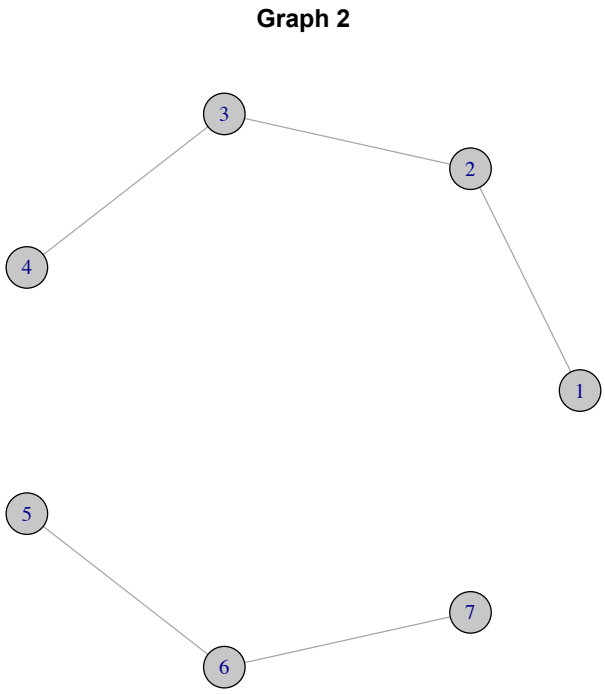

Figure 2.2: *Graph of example 2.*

x_4	0	1
$p(x_4\|X_3=0)$	0.7	0.3
$p(x_4\|X_3=1)$	0.2	0.8

x_5	0	1
$p(x_5\|X_6=0)$	0.3	0.7
$p(x_5\|X_6=1)$	0.8	0.2

x_7	0	1
$p(x_7\|X_6=0)$	0.5	0.5
$p(x_7\|X_6=1)$	0.3	0.7

c) Conditional probability of $X_1|X_2 = x_2$:

x_1	0	1
$p(x_1\|X_2=0)$	0.3	0.7
$p(x_1\|X_2=1)$	0.6	0.4

In this example the probability of all variables be equal to 0 is:

$$\begin{aligned} & p(X_1=0, X_2=0, X_3=0, X_4=0, X_5=0, X_6=0, X_7=0) \\ = \; & p(X_1=0|X_2=0)p(X_2=0|X_3=0)p(X_4=0|X_3=0)p(X_3=0) \\ & p(X_5=0|X_6=0)p(X_7=0|X_6=0)p(X_6=0) \\ = \; & 0.3 \times 0.4 \times 0.7 \times 0.4 \times 0.3 \times 0.5 \times 0.6 = 0.003024 \end{aligned}$$

2.3.3 Example 3

This third model has 12 random variables that assume values in $A = \{0, 1, 2\}$. And we assume that this model is a homogeneous Markov chain (represented in Figure 2.3), so the model's joint probability can be written as:

$$p(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}) = p(x_1) \prod_{i=1}^{11} p(x_{i+1}|x_i) \tag{2.14}$$

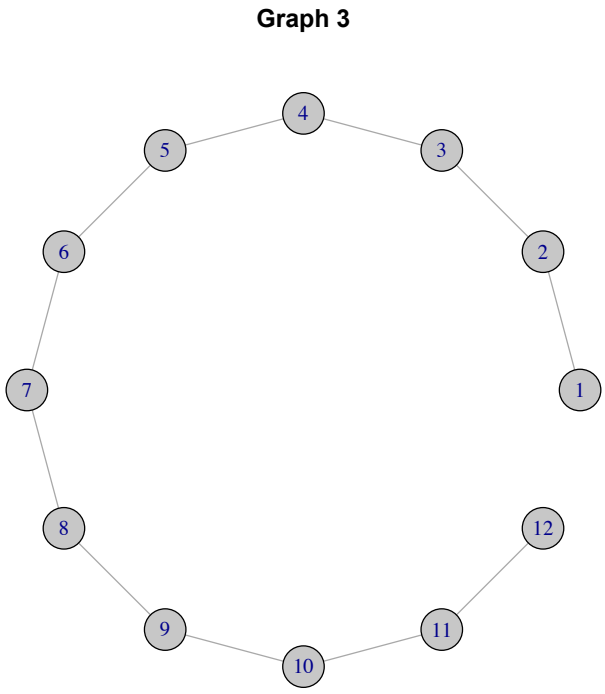

Figure 2.3: *Graph of example 3.*

And the chosen initial distribution and transition matrix of the Markov chain are:

$$p(x_1) = \begin{pmatrix} 0.3 & 0.6 & 0.1 \end{pmatrix} \qquad \text{e} \qquad \pi = \begin{bmatrix} 0.2 & 0.5 & 0.3 \\ 0.7 & 0.2 & 0.1 \\ 0.4 & 0.3 & 0.3 \end{bmatrix}.$$

For this third example the probability of all variables be equal to 0 is:

$$\begin{aligned} & p(X_1 = 0, X_2 = 0, X_3 = 0, X_4 = 0, X_5 = 0, X_6 = 0, X_7 = 0, X_8 = 0, X_9 = 0, \\ & X_{10} = 0, X_{11} = 0, X_{12} = 0) \\ = \; & p(X_1 = 0) \prod_{i=1}^{11} p(X_{i+1} = 0|X_i = 0) \\ = \; & 0.3 \times 0.2^{11} = 6,144 \times 10^{-9} \end{aligned} \tag{2.15}$$

2.3.4 Example 4

In this last example we have 10 random variables that assume values in $A = \{0, 1, 2\}$, and its joint probability is given by

$$\begin{aligned} & p(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}) \\ = \; & p(x_1|x_2)p(x_2|x_4)p(x_3|x_4)p(x_5|x_4)p(x_6|x_4)p(x_4) \\ & p(x_7|x_{10})p(x_8|x_9, x_{10})p(x_9|x_{10})p(x_{10}) \end{aligned} \tag{2.16}$$

The dependence structure can be seen in Figure 2.4, and the theoretical values chosen for the probabilities are given below:

a) Marginal distributions of X_4 and X_{10}:

x_4	0	1	2
$p(x_4)$	0.3	0.4	0.3

x_{10}	0	1	2
$p(x_{10})$	0.4	0.1	0.5

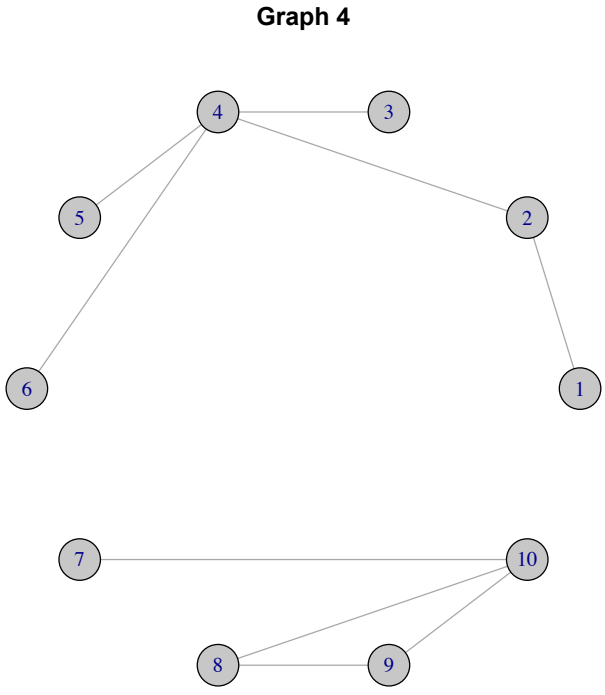

Figure 2.4: *Graph of example 4.*

b) Conditional probabilities of $X_i|X_4 = x_4$, $i = 2, 3, 5, 6$ and $X_j|X_{10} = x_{10}$, $j = 7, 9$:

x_2	0	1	2
$p(x_2\|X_4 = 0)$	0.4	0.4	0.2
$p(x_2\|X_4 = 1)$	0.5	0.2	0.3
$p(x_2\|X_4 = 2)$	0.5	0.3	0.2

x_3	0	1	2
$p(x_3\|X_4=0)$	0.2	0.6	0.2
$p(x_3\|X_4=1)$	0.3	0.3	0.4
$p(x_3\|X_4=2)$	0.4	0.2	0.4

x_5	0	1	2
$p(x_5\|X_4=0)$	0.7	0.2	0.1
$p(x_5\|X_4=1)$	0.3	0.5	0.2
$p(x_5\|X_4=2)$	0.1	0.5	0.4

x_6	0	1	2
$p(x_6\|X_4=0)$	0.3	0.2	0.5
$p(x_6\|X_4=1)$	0.6	0.2	0.2
$p(x_6\|X_4=2)$	0.4	0.3	0.3

x_7	0	1	2
$p(x_7\|X_{10}=0)$	0.3	0.3	0.4
$p(x_7\|X_{10}=1)$	0.4	0.2	0.4
$p(x_7\|X_{10}=2)$	0.5	0.3	0.2

x_9	0	1	2
$p(x_9\|X_{10}=0)$	0.4	0.4	0.2
$p(x_9\|X_{10}=1)$	0.3	0.4	0.3
$p(x_9\|X_{10}=2)$	0.5	0.2	0.3

c) Conditional probability of $X_1|X_2 = x_2$:

x_1	0	1	2
$p(x_1\|X_2=0)$	0.2	0.3	0.5
$p(x_1\|X_2=1)$	0.4	0.3	0.3
$p(x_1\|X_2=2)$	0.3	0.6	0.1

d) Conditional probability of $X_8|X_9 = x_9, X_{10} = x_{10}$:

x_8	0	1	2
$p(x_8\|X_9=0,X_{10}=0)$	0.75	0	0.25
$p(x_8\|X_9=1,X_{10}=0)$	0.25	0.5	0.25
$p(x_8\|X_9=2,X_{10}=0)$	0.5	0.5	0
$p(x_8\|X_9=0,X_{10}=1)$	0	2/3	1/3
$p(x_8\|X_9=1,X_{10}=1)$	0.5	0.25	0.25
$p(x_8\|X_9=2,X_{10}=1)$	1/3	1/3	1/3
$p(x_8\|X_9=0,X_{10}=2)$	0.2	0.4	0.4
$p(x_8\|X_9=1,X_{10}=2)$	0.5	0	0.5
$p(x_8\|X_9=2,X_{10}=2)$	1/3	1/3	1/3

For this last example the probability of all variables be equal to 0 is:

$$\begin{aligned} & p(X_1=0,X_2=0,X_3=0,X_4=0,X_5=0,X_6=0,X_7=0,X_8=0,X_9=0,X_{10}=0) \\ = \; & p(X_1=0|X_2=0)p(X_2=0|X_4=0)p(X_3=0|X_4=0)p(X_5=0|X_4=0) \\ & p(X_6=0|X_4=0)p(X_4=0)p(X_7=0|X_{10}=0)p(X_8=0|X_9=0,X_{10}=0) \\ & p(X_9=0|X_{10}=0)p(X_{10}=0) \\ = \; & 0.2\times 0.4\times 0.2\times 0.7\times 0.3\times 0.3\times 0.3\times 0.75\times 0.4\times 0.4 = 3.6288\times 10^{-5} \end{aligned} \tag{2.17}$$

As in the first example, here we do not have all marginal distributions strictly positve (look at the probabilities in item d)).

The next section presents how to estimate the probabilities in a Markov random field over A^V with graph G when the graph structure is known.

2.4 Estimation

Suppose we observe an independent sample with size n of the Markov random field $\{X_v : v \in V\}$. Denote by x_v^i the value obtained at the vertex v on the i-th observation of the sample. For each vertex $v \in V$ we will estimate the neighborhood $\mathrm{ne}(v)$ and the conditional probabilities given by (2.8).

Given a vertex $v \in V$ and a set $W \subset V$ not containing v, the operator $N(a, a_W)$ will denote the number of occurrences of the event

$$\{X_v = a\} \cap \{X_W = a_W\}$$

in the sample. That is, for a sample with size n

$$N(a, a_W) \;=\; \sum_{i=1}^{n} \mathbf{1}\{x_v^i = a\,,\; x_W^i = a_W\}\,.$$

The maximum likelihood estimator of the conditional distribution (2.8) is given by

$$\hat{p}(a|a_W) = \frac{N(a, a_W)}{N(a_W)}, \qquad \text{for } a \in A\,, \tag{2.18}$$

where $N(a_W) = \sum_{a \in A} N(a, a_W)$. If $N(a_W) = 0$ we adopt the convention $\hat{p}(a|a_W) = 1/|A|$ for all $a \in A$. Therefore, the maximum likelihood of the conditional distribution of X_v given X_W is

$$\hat{\mathbb{P}}(x_v^{(1:n)}|x_W^{(1:n)}) = \prod_{a_W \in A^W} \prod_{a \in A} \hat{p}(a|a_W)^{N(a,a_W)}\,. \tag{2.19}$$

In order to estimate the neighborhood $\mathrm{ne}(v)$ of the vertex $v \in V$ we propose a penalized maximum conditional likelihood criterion, similar to the Bayesian Information Criterion (BIC) of Schwarz (1978). In the following section we present the concept of model selection and define the estimators of the neighborhood.

2.5 Model selection

Model selection can simply be translated as the task of selecting a statistical model from a set of candidate models given a sample. Since there is the set of candidate models it is necessary to choose the best model among them. However, the meaning of the best can be controversial. A good model selection technique should consider the good fit of the data and the complexity of the model, finding a balance between these two criteria (Wallace and Boulton (1968) and Hastie et al. (2009)). After all, it is obvious that more complex models can incorporate more information coming from the sample, but the added parameters may not represent anything useful to the problem and can increase the variance.

In our case, we look at each vertex of the graph and seek, among a set of possible candidate subsets, the one that has the best fit penalized by the cardinality of this subset. The set $V \setminus \{v\}$ contains all the possible neighbors of v, note that we have a total of $\sum_{i=0}^{|V\setminus\{v\}|} \binom{|V\setminus\{v\}|}{i} = 2^{|V\setminus\{v\}|} = 2^{|V|-1}$ subsets, that is, we have $2^{|V|-1}$ possible models for the neighborhood of the vertex v. Also, note that for the subset W of neighbors and an alphabet A, the number of model parameters is

equal to $(|A|-1)|A|^{|W|}$ and, if the number of neighbors increase in one unit, the number of model parameters increase in

$$(|A|-1)|A|^{|W|+1} - (|A|-1)|A|^{|W|} = |A|^{|W|+2} - 2|A|^{|W|+1} + |A|^{|W|}.$$

The neighborhood estimator proposed in this thesis uses methods of penalized maximum conditional likelihood similar to the Bayesian Information Criterion (BIC), as already said. In the sequel we define the criterion and prove its consistency.

Definition 6. *Given a constant $c > 0$, the empirical neighborhood of the vertex v is the set of indices $\widehat{ne}(v)$ given by*

$$\widehat{ne}(v) \;=\; \underset{W \subset V \setminus \{v\}}{\arg\max} \left\{ \log \hat{\mathbb{P}}(x_v^{(1:n)} | x_W^{(1:n)}) - c\,|A|^{|W|} \log_{|A|} n \right\}, \tag{2.20}$$

where $V \setminus \{v\}$ is the set of all possible neighbors of vertex v.

We prove the following consistency result for the neighborhood estimator.

Theorem 7. *For any $v \in V$ and any $c > 0$, the estimator given by (2.20) satisfies $\widehat{ne}(v) = ne(v)$ eventually almost surely as $n \to \infty$.*

Our main goal in this thesis is to estimate the graph G from a finite sample. To do this we can estimate the neighborhood of each node and reconstruct a graph based on the set of neighborhoods. Based on the neighborhood estimator (2.20), we can construct an estimator of the graph G by defining the set of edges

$$\widehat{E}^{-} = \{(v,w) \in V \times V \colon v \in \widehat{\mathrm{ne}}(w) \text{ and } w \in \widehat{\mathrm{ne}}(v)\}.$$

The estimated graph will be the pair $\widehat{G}^{-} = (V, \widehat{E}^{-})$. In the same way, if we want to be less conservative we can define

$$\widehat{E}^{+} = \{(v,w) \in V \times V \colon v \in \widehat{\mathrm{ne}}(w) \text{ or } w \in \widehat{\mathrm{ne}}(v)\}$$

and the estimated graph will be the pair $\widehat{G}^{+} = (V, \widehat{E}^{+})$.

Corollary 8. *For all finite V we have $\widehat{G}^{-} = \widehat{G}^{+} = G$ eventually almost surely as $n \to \infty$.*

Before presenting the proof of Theorem 7 and Corollary 8 we prove a proposition that will be used in the proof of those results. And, from now on we simply write $\log n$ for the logarithm in base $|A|$.

Proposition 9. *For any $\delta > 0$ and any pair $(a, a_W) \in A^{W+1}$ with $W \subset V \setminus \{v\}$ we have*

$$|\hat{p}(a|a_W) - p(a|a_W)| < \sqrt{\frac{\delta \log n}{N(a_W)}} \tag{2.21}$$

eventually almost surely as $n \to \infty$.

Proof. Define, for a fixed $(a, a_W) \in A^{W+1}$, the random variables

$$Y_i = \mathbf{1}\{x_v^{(i)} = a, x_W^{(i)} = a_W\} - p(a|a_W)\mathbf{1}\{x_W^{(i)} = a_W\}, \; i = 1, 2, \ldots, n$$

and

$$Z_n = \sum_{i=1}^{n} Y_i = N(a, a_W) - p(a|a_W)N(a_W).$$

The variables $\{Y_i : i = 1, 2, \ldots, n\}$ are independent and identically distributed. Note that

$$\begin{aligned}
&\mathbb{P}(Y_i = 1 - p(a|a_W)) = p(a, a_W),\\
&\mathbb{P}(Y_i = -p(a|a_W)) = \sum_{b \neq a} p(b, a_W) = p(a_W) - p(a, a_W),\\
&\mathbb{P}(Y_i = 0) = 1 - p(a_W),
\end{aligned}$$

where $p(a_W) = \sum_{a' \in A} p(a', a_W)$. So, we have that

$$\begin{aligned}
\mathbb{E}(Y_i) &= \{1 - p(a|a_W)\}\, p(a, a_W) - p(a|a_W)\, \{p(a_W) - p(a, a_W)\}\\
&= p(a, a_W) - \frac{p(a, a_W)}{p(a_W)} p(a, a_W) - \frac{p(a, a_W)}{p(a_W)} p(a_W)\\
&\quad + \frac{p(a, a_W)}{p(a_W)} p(a, a_W) = 0
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}(Y_i^2) &= \left\{1 - 2\frac{p(a, a_W)}{p(a_W)} + \frac{p(a, a_W)^2}{p(a_W)^2}\right\} p(a, a_W)\\
&\quad + \frac{p(a, a_W)^2}{p(a_W)^2}\, \{p(a_w) - p(a, a_W)\}\\
&= \frac{p(a, a_W)}{p(a_W)}\, \{p(a_W) - p(a, a_W)\}\\
&= p(a|a_W)\, \{1 - p(a|a_W)\}\, p(a_W)\\
&\leq \frac{1}{4} p(a_W) \qquad (2.22)
\end{aligned}$$

Now, by the Law of the Iterated Logarithm (Appendix A, 26) we have that for any $\epsilon > 0$,

$$|Z_n| < (1 + \epsilon)\frac{p(a_W)}{4}\sqrt{2n \log\log n} = (1 + \epsilon)\frac{1}{4}\sqrt{2p(a_W)^2 n \log\log n}$$

eventually almost surely as $n \to \infty$. In particular we have

$$|Z_n| < \sqrt{2p(a_W)^2 n \log\log n}$$

eventually almost surely as $n \to \infty$. Dividing both sides of the inequality by $N(a_W)$ we obtain that

$$|\hat{p}(a|a_W) - p(a|a_W)| < \sqrt{\frac{2p(a_W)^2 n \log\log n}{N(a_W)^2}}$$

eventually almost surely as $n \to \infty$. By the Strong Law of Large Numbers (Appendix A, 27) $n/N(a_W) \to 1/p(a_W)$ almost surely, therefore we have

$$|\hat{p}(a|x_W) - p(a|a_W)| < \sqrt{\frac{2p(a_W) \log\log n}{N(a_W)}}$$

eventually almost surely as $n \to \infty$. Now, for any $\delta > 0$ we have

$$2p(a_W) \log\log n < \delta \log n$$

eventually as $n \to \infty$. Ergo

$$|\hat{p}(a|a_W) - p(a|a_W)| < \sqrt{\frac{\delta \log n}{N(a_W)}}.$$

□

Now it is presented the proof of Theorem 7.

Proof of Theorem 7. For $v \in V$ and $W \subset V \setminus \{v\}$ denote by

$$\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;=\; \log \hat{\mathbb{P}}(x_v^{(1:n)}|x_W^{(1:n)}) - c|A|^{|W|} \log n$$

where

$$\hat{\mathbb{P}}(x_v^{(1:n)}|x_W^{(1:n)}) = \prod_{x_W \in A^{|W|}} \prod_{a \in A} \hat{p}(a|a_W)^{N(a,a_W)}\,.$$

We will show that for all $W \subset V \setminus \{v\}$, $W \neq \mathrm{ne}(v)$

$$\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely as $n \to \infty$. As the set of all such W is finite this implies the statement of the theorem. Then let $v \in V$ and let $W \subset V \setminus \{v\}$, with $W \neq \mathrm{ne}(v)$. We have the following possibilities

(a) $\mathrm{ne}(v) \subset W$;

(b) $W \subset \mathrm{ne}(v)$;

(c) neither $\mathrm{ne}(v) \subset W$ nor $W \subset \mathrm{ne}(v)$.

By ease of exposition we will divide the proof in these three cases.
(a) We have to prove that for all $W \supset \mathrm{ne}(v)$

$$\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely $n \to \infty$.

Observe that

$$\begin{aligned}\mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) - \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;=& \\ c\left(|A|^{|W|} - |A|^{|\mathrm{ne}(v)|}\right)\log n \;-& \sum_{(a,a_W)\in A^{W+1}} N(a,a_W)\log\frac{\hat{p}(a|a_W)}{\hat{p}(a|a_{\mathrm{ne}(v)})}\,. \end{aligned} \tag{2.23}$$

As these empirical probabilities are the maximum likelihood estimators, $\mathrm{ne}(v) \subset W$ and keeping in mind that $\mathrm{ne}(v)$ by definition is the smallest Markov neighborhood of v, we have that

$$\begin{aligned}\sum_{(a,a_W)\in A^{W+1}} N(a,a_W)\log\hat{p}(a|a_{\mathrm{ne}(v)}) \;&\geq \sum_{(a,a_W)\in A^{W+1}} N(a,a_W)\log p(a|a_{\mathrm{ne}(v)}) \\ &= \sum_{(a,a_W)\in A^{W+1}} N(a,a_W)\log p(a|a_W)\,.\end{aligned}$$

Therefore, (2.23) can be lower-bounded by

$$c\left(1 - \frac{1}{|A|}\right)|A|^{|W|}\log n \;- \sum_{(a,a_W)\in A^{W+1}} N(a,a_W)\log\frac{\hat{p}(a|a_W)}{p(a|a_W)}\,.$$

Now, observe that

$$\sum_{(a,a_W)\in A^{W+1}} N(a,a_W)\log\frac{\hat{p}(a|a_W)}{p(a|a_W)} = \sum_{a_W\in A^W} N(a_W)D(\hat{p}(\cdot|a_W)\,;\,p(\cdot|a_W))\,,$$

where D denotes the *Küllback-Leibler divergence* (see Definition 22 in Appendix A). Therefore we have, by Lemma 24 (in Appendix A) and Proposition 9, that for any $\delta > 0$

$$\begin{aligned}
&\sum_{a_W\in A^W} N(a_W)D(\hat{p}(\cdot|a_W)\,;\,p(\cdot|a_W)) \\
&\qquad\leq \sum_{a_W\in A^W} N(a_W)\sum_{a\in A}\frac{[\,\hat{p}(a|a_W)-p(a|a_W)\,]^2}{p(a|a_W)} \\
&\qquad\leq \sum_{a_W\in A^W} N(a_W)\sum_{a\in A}\frac{\delta\log n}{N(a_W)p(a|a_W)} \\
&\qquad\leq \frac{\delta|A|^{|W|+1}\log n}{p_{\min}}\,,
\end{aligned}$$

eventually almost surely as $n\to\infty$, where

$$p_{\min} \;=\; \min\{\,p(a|a_W)\colon\; p(a|a_W)>0,\, a\in A,\, a_W\in A^W\}.$$

Then, if we take $c > \frac{\delta|A|^2}{p_{\min}(|A|-1)}$, we have that eventually almost surely as $n\to\infty$

$$\mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) \;>\; \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}).$$

This completes the proof of part (a).
(b) We have to prove that for all $W\subset \mathrm{ne}(v)$

$$\mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) \;>\; \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)})$$

eventually almost surely as $n\to\infty$.

In this case we have that

$$\begin{aligned}
&\mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) - \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;= \\
&\qquad\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} N(a,a_{\mathrm{ne}(v)})\log\frac{\hat{p}(a|a_{\mathrm{ne}(v)})}{\hat{p}(a|a_W)} - c\,(|A|^{|\mathrm{ne}(v)|}-|A|^{|W|})\log n \\
&\quad= n\left[\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} \frac{N(a,a_{\mathrm{ne}(v)})}{n}\log\frac{\hat{p}(a|a_{\mathrm{ne}(v)})}{\hat{p}(a|a_W)} - c\,(|A|^{|\mathrm{ne}(v)|}-|A|^{|W|})\frac{\log n}{n}\right].
\end{aligned}$$

By the Strong Law of Large Numbers (Appendix A, Theorem 26) we have that

$$\begin{aligned}
&\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} \frac{N(a,a_{\mathrm{ne}(v)})}{n}\log\frac{\hat{p}(a|a_{\mathrm{ne}(v)})}{\hat{p}(a|a_W)} \\
&\qquad\longrightarrow \sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} p(a,a_{\mathrm{ne}(v)})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \qquad (2.24)
\end{aligned}$$

almost surely as $n \to \infty$. On the other hand,

$$c\left(|A|^{|\mathrm{ne}(v)|} - |A|^{|W|}\right)\frac{\log n}{n} \;\longrightarrow\; 0$$

when $n \to \infty$. Now, note that we can rewrite the right-hand side of (2.24) by

$$\begin{aligned}
&\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} p(a,a_{\mathrm{ne}(v)})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \\
&\qquad = \sum_{x_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}} p(a_{\mathrm{ne}(v)})\sum_{a\in A} p(a|a_{\mathrm{ne}(v)})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \\
&\qquad = \sum_{a_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}} p(a_{\mathrm{ne}(v)})\; D(p(.|a_{\mathrm{ne}(v)})\,;\,p(.|a_W))\,.
\end{aligned}$$

By Lemma 23 (in Appendix A) and the minimality of $\mathrm{ne}(v)$ (see (2.3) and Lemmas 3 and 4) we must have $D(p(.|a_{\mathrm{ne}(v)})\,;\,p(.|a_W)) > 0$ for at least one $a_{\mathrm{ne}(v)}$, so

$$\sum_{a_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}} p(a_{\mathrm{ne}(v)})\; D(p(.|x_{\mathrm{ne}(v)})\,;\,p(.|a_W)) > 0.$$

Therefore

$$\mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) \;>\; \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)})$$

eventually almost surely as $n \to \infty$, completing the proof of case (b).
(c) In this case, let W such that neither $W \subset \mathrm{ne}(v)$ nor $\mathrm{ne}(v) \subset W$, and consider $W' = W \cup \mathrm{ne}(v)$. We will prove that

$$\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{W'}^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely as $n \to \infty$.

As $W' \supset \mathrm{ne}(v)$, the second inequality is valid by case (a) proved above, so we just have to prove the first inequality. Note that we have

$$\begin{aligned}
&\mathrm{PML}(x_v^{(1:n)}|x_{W'}^{(1:n)}) - \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;= \\
&\qquad \sum_{(a,a_{W'})\in A^{W'+1}} N(a,a_{W'})\log\frac{\hat{p}(a|a_{W'})}{\hat{p}(a|a_W)} - c\left(|A|^{|W'|} - |A|^{|W|}\right)\log n \\
&= n\left[\sum_{(a,a_{W'})\in A^{W'+1}} \frac{N(a,a_{W'})}{n}\log\frac{\hat{p}(a|a_{W'})}{\hat{p}(a|a_W)} - c\left(|A|^{|W'|} - |A|^{|W|}\right)\frac{\log n}{n}\right].
\end{aligned}$$

By the Strong Law of Large Numbers (Appendix A, Theorem 26) we have that the first term in the brackets

$$\begin{aligned}
&\sum_{(a,a_{W'})\in A^{W'+1}} \frac{N(a,a_{W'})}{n}\log\frac{\hat{p}(a|a_{W'})}{\hat{p}(a|a_W)} \\
&\qquad\qquad \longrightarrow \sum_{(a,a_{W'})\in A^{W'+1}} p(a,a_{W'})\log\frac{p(a|a_{W'})}{p(a|a_W)}
\end{aligned}$$

almost surely as $n \to \infty$. On the other hand the second term

$$c\left(|A|^{|W'|} - |A|^{|W|}\right)\frac{\log n}{n} \;\longrightarrow\; 0$$

when $n \to \infty$. Now, note that

$$\sum_{(a,a_{W'})\in A^{W'+1}} p(a,a_{W'}) \log \frac{p(a|a_{W'})}{p(a|a_W)}$$
$$= \sum_{a_{W'}\in A^{W'}} p(a_{W'}) \sum_{a\in A} p(a|a_{W'}) \log \frac{p(a|a_{W'})}{p(a|a_W)} =$$
$$= \sum_{a_{W'}\in A^{W'}} p(a_{W'})\; D(p(\cdot|a_{W'})\,;\, p(\cdot|a_W))\,.$$

By Lemma 23 (in Appendix A), $D(p(.|a_{W'})\,;\, p(.|a_W)) \geq 0$ for all $a_{W'} \in A^{W'}$. If $D(p(.|a_{W'})\,;\, p(.|a_W)) = 0$ for all $a_{W'} \in A^{W'}$ then $p(a|a_{W'}) = p(a|a_W)$ for all $a_{W'} \in A^{W'}$ and all $a \in A$. On the other hand, as $\mathrm{ne}(v)$ is a Markov neighborhood and $\mathrm{ne}(v) \subset W'$ we have that

$$p(a|a_{W'}) = p(a|a_{\mathrm{ne}(v)}) \text{ for all } a_{W'} \in A^{W'} \text{ and all } a \in A.$$

Therefore we have that

$$p(a|a_W) = p(a|a_{\mathrm{ne}(v)}) \text{ for all } x_W \in A^W \text{ and all } a \in A.$$

By Lemma 4, W is a Markov neighborhood, i.e. $W \supset \mathrm{ne}(v)$, which contradicts the hypothesis in this case. So, $D(p(.|a_{W'})\,;\, p(.|a_W)) > 0$ for at least one $a_{W'} \in A^{W'}$ and then

$$\sum_{a_{W'}\in A^{W'}} p(a_{W'})\; D(p(\cdot|a_{W'})\,;\, p(\cdot|a_W)) > 0.$$

Therefore

$$\mathrm{PML}(x_v^{(1:n)}|x_{W'}^{(1:n)}) \;>\; \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)})$$

eventually almost surely as $n \to \infty$, finishing the proof of case (c).

With that, we proved that for any $W \subset V \setminus \{v\}$

$$\mathrm{PML}_{\mathrm{ne}(v)}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) \;>\; \mathrm{PML}_W(x_v^{(1:n)}|x_W^{(1:n)})$$

eventually almost surely as $n \to \infty$, and as the set of possible subsets W is finite we have

$$\mathrm{PML}_{\mathrm{ne}(v)}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) \;>\; \max_{W\subset V\setminus\{v\}, W\neq \mathrm{ne}(v)} \{\, \mathrm{PML}_W(x_v^{(1:n)}|x_W^{(1:n)}) \,\}$$

eventually almost surely as $n \to \infty$; that is, the estimator given in 2.20 satisfies $\widehat{\mathrm{ne}}(v) = \mathrm{ne}(v)$ eventually almost surely as $n \to \infty$. □

Proof of Corollary 8. The proof of the corollary follows from Theorem 7, by noting that V is finite. That is, for each vertex $v \in V$ we have that $\widehat{\mathrm{ne}}(v) = \mathrm{ne}(v)$ eventually almost surely as $n \to \infty$, i.e., for each vertex $v \in V$ there exists a sample size $n(v)$ large enough such that with probability one, $\widehat{\mathrm{ne}}(v) = \mathrm{ne}(v)$ for all $n \geq n(v)$. As V is finite we can take

$$n^* = \max_{v\in V}\{\, n(v) \,\}\,,$$

then with probability one $\cap_{v\in V}\{\widehat{\mathrm{ne}}(v) = \mathrm{ne}(v)\}$ for all $n \geq n^*$ and this implies that, with probability one, $\widehat{G}^- = \widehat{G}^+ = G$ for all $n \geq n^*$. □

Chapter 3

Simulations

This chapter presents some results for the examples in Chapter 2 througth simulations. First let's see how to generate samples from these examples, then how to use the estimation program. Next the results of the estimations will be evaluated from different perspectives: considering different values for the constant c of the estimator and sample sizes.

3.1 Generating samples

For each example there is a different way to generate a sample from the Markov random field on A^V with graph G. The simplest way to do that is using the factorized joint probability, i.e., using the conditional distributions insted of using the joing probability. Another knowledge required is how to generate a random variable, one way to do that is using the inverse transform method (see Rizzo (2007)). Now let's see each one of the algorithms, and the programs in R language[1] that performs these algorithms are in Appendix B, Section B.1.

3.1.1 Example 1

This first example had its joint probability factorized like this (see 2.9):

$$p(x_1, x_2, x_3, x_4, x_5) = p(x_2|x_1, x_3)p(x_1|x_3)p(x_4|x_3)p(x_5|x_3)p(x_3)$$

So, to generate a sample from this Markov random field we follow this steps:

1. Generate the random variable X_3 from its marginal distribution;
2. Given $X_3 = x_3$ in the first step, generate the random variables X_1, X_4 and X_5 from their conditional distributions $X_i|X_3 = x_3$, $i = 1, 4, 5$;
3. Given $X_3 = x_3$ (step 1) and $X_1 = x_1$ (step 2), generate the random variable X_2 from its conditional distribution $X_2|X_1 = x_1, X_3 = x_3$.

If it is wanted to generate a sample with size n, the steps 1-3 have to be repeated n times.

3.1.2 Example 2

Now, this example has its joint probability factorized like this (see 2.13):

$$p(x_1, x_2, x_3, x_4, x_5, x_6, x_7) = p(x_1|x_2)p(x_2|x_3)p(x_4|x_3)p(x_3)p(x_5|x_6)p(x_7|x_6)p(x_6)$$

Then, the steps to generating a sample from this Markov random field are the following:

1. Generate the random variables X_3 and X_6 from their marginal distributions;

[1] www.r-project.org

2. Given $X_3 = x_3$ in the first step, generate the random variables X_1, X_2 and X_4 from their conditional distributions $X_i|X_3 = x_3$, $i = 1, 2, 4$, and given $X_6 = x_6$ (step 1) generate the random variables X_5 and X_7 from their conditional distributions $X_j|X_6 = x_6$, $j = 5, 7$.

To generate a sample with size n, just repeat the steps 1 and 2 n times.

3.1.3 Example 3

This example has its joint probability factorized like this (see 2.14):

$$p(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}) = p(x_1) \prod_{i=1}^{11} p(x_{i+1}|x_i)$$

Keeping in mind that this example represents a Markov chain with order 1 the steps to generate a sample from this model are:

1. Generate the random variable X_1 from the initial distribution;
2. Given $X_1 = x_1$ (step 1), generate the random variable X_2 through the transition matrix π of the Markov chain;
3. Given $X_2 = x_2$ (step 2), generate the random variable X_3 through the transition matrix π of the Markov chain;

...

12. Given $X_{11} = x_{11}$ (step 11), generate the random variable X_{12} through the transition matrix π of the Markov chain.

And then, to generate a sample with size n the steps 1-12 have to be repeated n times.

3.1.4 Example 4

This last example has its joint probability factorized like this (see 2.16):

$$\begin{aligned} & p(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}) \\ = \; & p(x_1|x_2)p(x_2|x_4)p(x_3|x_4)p(x_5|x_4)p(x_6|x_4)p(x_4)p(x_7|x_{10})p(x_8|x_9, x_{10})p(x_9|x_{10})p(x_{10}) \end{aligned}$$

In this case, the steps to generate a sample are:

1. Generate the random variables X_4 and X_{10} from their marginal distributions;
2. Given $X_4 = x_4$ (step 1), generate the random variables X_1, X_2, X_3 and X_5 from their conditional distributions $X_i|X_4 = x_4$, $i = 1, 2, 3, 5$, and given $X_{10} = x_{10}$ (step 1) generate the random variables X_7 and X_9 from their conditional distributions $X_j|X_{10} = x_{10}$, $j = 7, 9$;
3. Given $X_{10} = x_{10}$ (step 1) and $X_9 = x_9$ (step 2), generate the random variable X_8 from its conditional distribution $X_8|X_9 = x_9, X_{10} = x_{10}$.

Repeate the steps 1-3 n times to have a sample with size n.

3.2 How to use the estimation program

With the help of the programmers at NUMEC-USP[2] (Support Center for Stochastic Modeling and Complexity at Universtity of São Paulo), it was developed a program[3], in C language, that,

[2] http://www.numec.prp.usp.br/

[3] available in http://github.com/yoshiomori/neighborhoods.git

considering a sample, estimates the appropriate graph based on the estimator proposed in this thesis (Sections 2.4 and 2.5).

The program take into count informations:

a) the alphabet A;

b) the value c of the estimator constant;

c) the sample;

d) the choice if the estimator should be conservative or not;

e) and the adjacency matrix (simetric matrix with zeros and ones, and diagonal of zeros) that gives the maximal neighborhood that should be tested.

The last item in this list, the adjacency matrix, it is useful when you have a prior knowledge about the neighborhoods, that is, when the absence of some of the edges is known, decreasing the number of comparisons made by the algorithm. If this information is unknown the adjacency matrix should represent the complete graph, i.e., to find the neighborhood of a vertex the algorithm will test all possible combinantions of the remaining vertices.

So, the program reads a file with the alphabet A, the value of c, the sample (a $|V| \times n$ matrix) and the adjacency matrix. The following presents an example of this file for an alphabet $A = (2, 3)$, a $c = 1.5$, a sample of $X = (X_1, X_2, X_3)$ of size 10 and an adjacency matrix of a complet graph:

```
2 3

1.5

2 3 3 2 2 3 2 2 3 3
3 3 3 2 2 2 3 2 2 3
2 2 3 3 2 3 2 3 3 2

0 1 1
1 0 1
1 1 0
```

Given this file, the choice between a conservative or non conservative algorithm is made at the time of program execution, 0 (zero) represents the conservative choice and 1 (one) the non conservative one. And, as a result, the program gives two output files: a -neig.txt, that contains the adjacency matrix of the graph that was estimated; and a -prob.txt, that contains the estimates conditional probabilities of each vertex given its estimated neighborhood.

In the next section this program was used to evaluate the performance of the estimator proposed in this thesis, as well as used in Chapter 4 in the applications to real datasets.

3.3 Estimator performance

In this section the estimator's performance it is evaluated in three different ways. The first one considers samples of diferent sizes and estimate the graphs using different values of the constant c with the two approaches (conservative and non conservative). In the second one, for a fixed value of c, the underestimation and overestimation errors are estimated, making it possible to evaluate their behavior with the growth of sample size. The last evaluation uses ROC (Receiver Operating Characteristic) curves (see Fawcett (2006)) to compare the conservative and non conservative approaches, for different values of c and a fixed sample size.

3.3.1 Evaluating convergence of the graphs

Here just one sample of each sample size was generated to be used by the program with different values of c. There were generated sample of sizes: 100, 500, 1, 000, 5, 000 and 10, 000, and the different values of c used were: 0.25, 0.5, 1, 1.5 and 2.

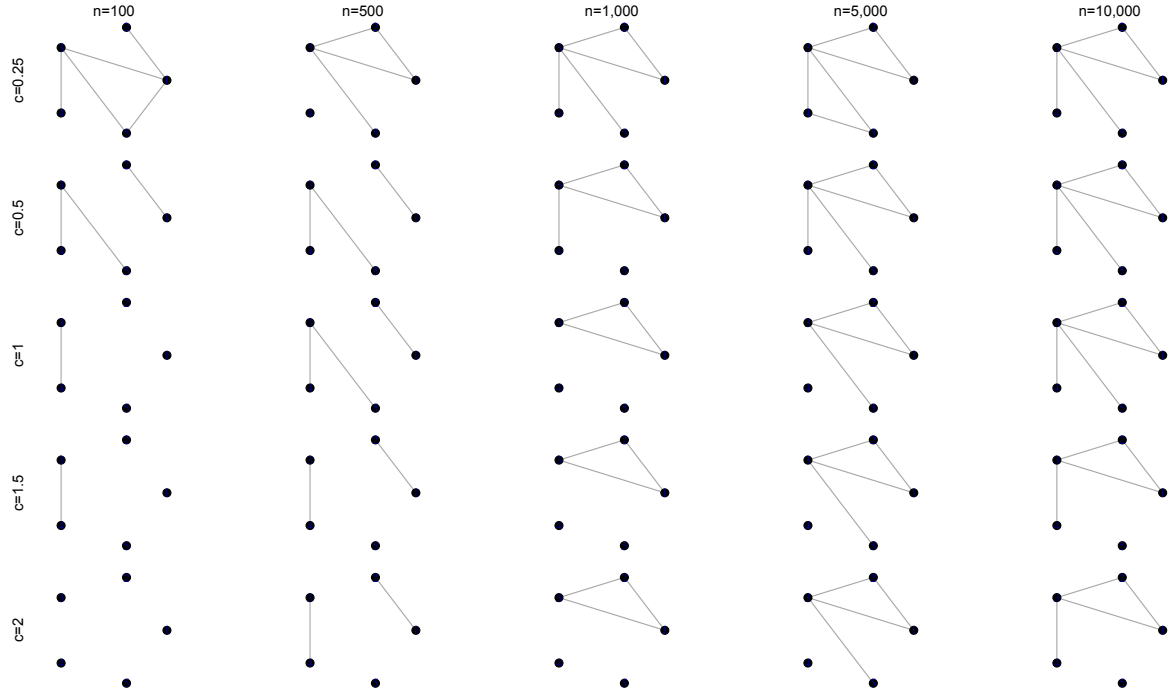

Figure 3.1: *Graph of example 1: Evolution of the estimated graph for different sample sizes and constant c considering the conservative option.*

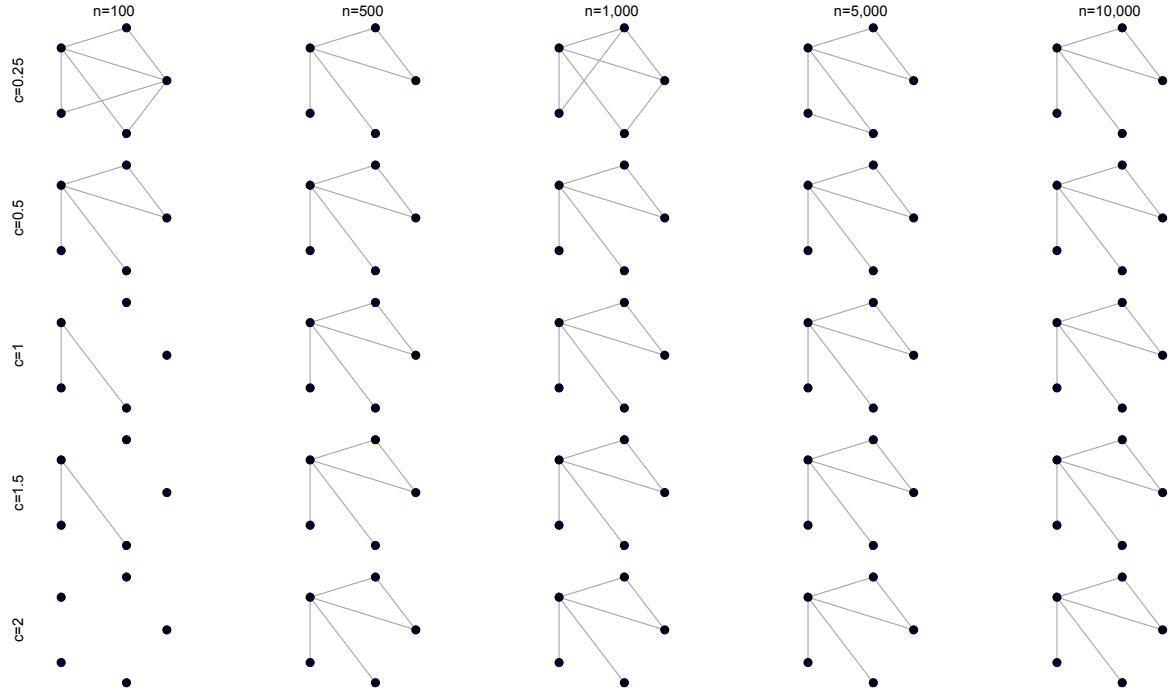

Figure 3.2: *Graph of example 1: Evolution of the estimated graph for different sample sizes and constant c considering the non conservative option.*

First they are presented the results of Example 1, Figures 3.1 and 3.2, considering the conservative and the non-conservative options, respectively.Compared to the results with the real graph,

Figure 2.1 in Chapter 2, with the simulation results it is possible to see that:

- For the sample with size $n = 100$, it wasn't possible to reconstruct the correct dependence structure of the graph for any of the values of c in the conservative option (Figure 3.1, first column), for $c = 0.5$, 1, 1.5 and 2 the graph was underestimated, and for $c = 0.25$ there is one missing edge and one spare edge. But for the non conservative option (Figure 3.2, first column) the correct dependence structue was found just for $c = 0.5$, for $c = 0.25$ there are two spares edges, and for the other values of c (the highest values) the graph was underestimated.

- For the sample with size $n = 500$, again it wasn't possible to reconstruct the correct dependence estructure of the graph for any of the values of c in the conservative option (Figure 3.1, second column), but in this case for all values of c the graph was underestimated. On the other hand, for the conservative option (Figure 3.2, second column) the graph was correctly estimated for all values of c.

- For the sample with size $n = 1,000$, in the conservative option (Figure 3.1, third column) just for $c = 0.25$ the graph was correctly estimated, for the other values of c the graph was underestimated. The opposite occurred in the non conservative option (Figure 3.2, third column), here just for $c = 0.25$ the graph wasn't correctly estimated, it was overestimated with two spare edges, and for the other values of c the graph was correctly estimated.

- For the sample with size $n = 5,000$, in the conservative option (Figure 3.1, fourth column) the graph was overestimated for $c = 0.25$ with one spare edge, it was correctly estimated for $c = 0.5$, and it was uderestimated for the other ones. In the non conservative option (Figure 3.2, fourth column), like in the other option, the graph was overestimated for $c = 0.25$, but it was correctly estimated for all the other values of c.

- And for the sample with size $n = 10,000$, in the conservative option (Figure 3.1, fifth column) the graph was correctly estimated for $c = 0.25$, 0.5 and 1, and it was underestimated for $c = 1.5$ and 2. In the non conservative option (Figure 3.2, fifth column), the graph was correctly estimated for all values of c.

In general, for this example the non conservative approach was better then the conservative one, being able to properly estimate the dependece structure of the graph 19 of 25 times against 5 of 25 times.

Now the results for Example 2 are presented in Figures 3.3, the conservative option, and Figure 3.4, the non conservative option. Comparing these results with the correct dependence estructure of the graph in Figure 2.2 (Chapter 2) it is possible to observe that:

- For the sample with size $n = 100$ the conservative option (Figure 3.3, first column) and the non conservative option (Figure 3.4, first column) were not capable of reconstructing the true graph structure. In the conservative option for $c = 0.25$ there is two spare edges and three missing edges, for the other values of c there is just missing edges. In the non conservative option for $c = 0.25$ there is two spare edges and two missing edges, for $c = 0.5$ there is two spare edges and three missing ones, and for the other values there are just missing edges.

- For the sample with size $n = 500$ again none of the options were able of reconstructing the true graph structure. In the conservative option (Figure 3.3, second column) the graph was underestimated for all values of c. In the non conservative option (Figure 3.4, second column) the graph was not properly estimated for any value of c, for $c = 0.25$ and 0.5 there is one spare edge and two missing edges, for the other values of c there are just missing edges.

- For the sample with size $n = 1,000$, in the conservative option (Figure 3.3, third column) the graph was underestimated for all values of c. But in the non conservative option (Figure 3.4, third column) the graph was correctly estimated only for $c = 0.25$, for the other values of c the graph was underestimated.

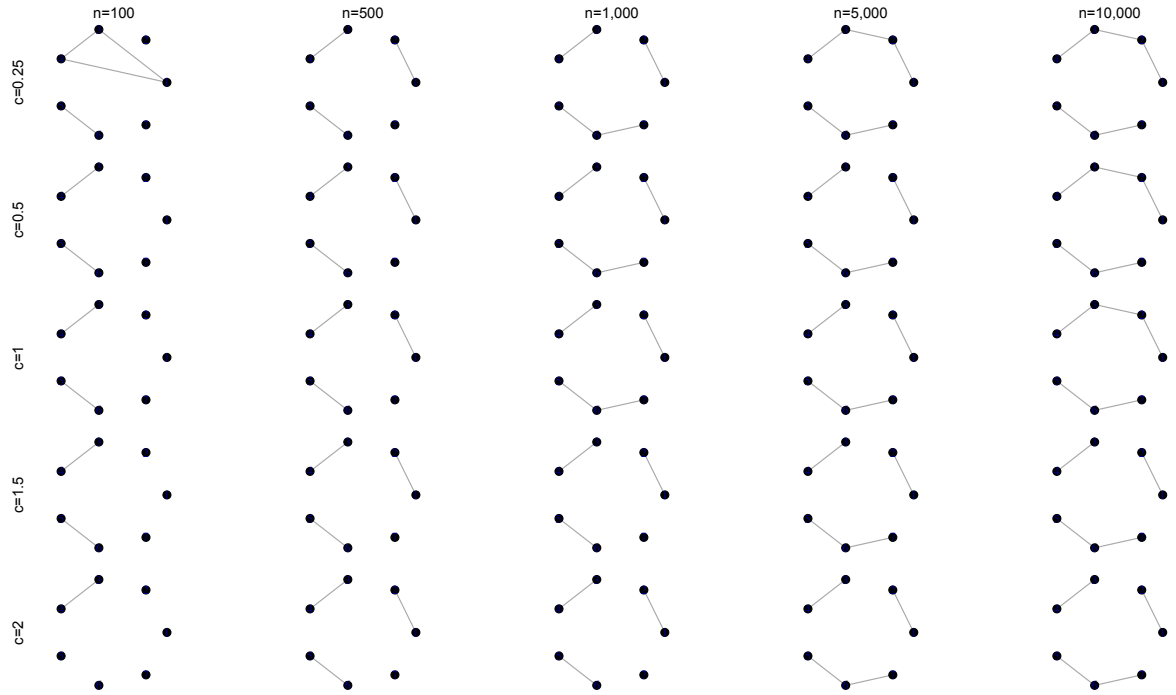

Figure 3.3: *Graph of example 2: Evolution of the estimated graph for different sample sizes and constant c considering the conservative option.*

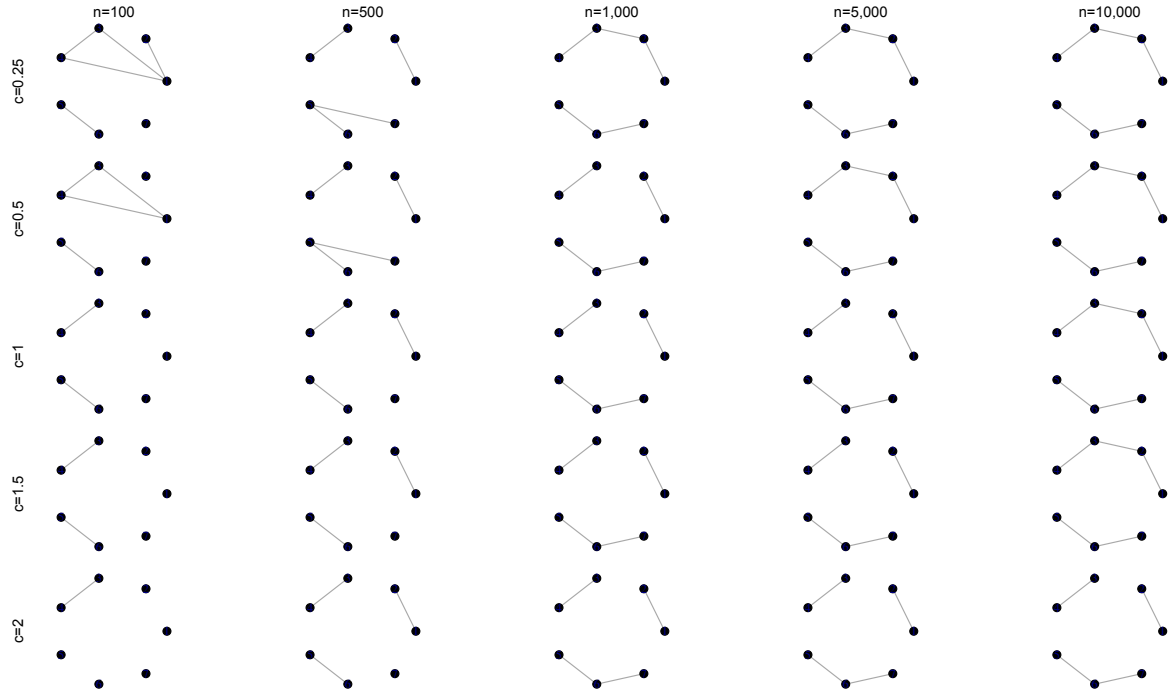

Figure 3.4: *Graph of example 2: Evolution of the estimated graph for different sample sizes and constant c considering the non conservative option.*

- For the sample with size $n = 5,000$, in the conservative option (Figure 3.3, fourth column) the graph was correctly estimated for $c = 0.25$, for the other values of c the graph was underestimated. In the non conservative option (Figure 3.4, fourth column) the graph was correctly estimated for $c = 0.25$ and 0.5, and it was underestimated for the other values of c.

- For the sample with size $n = 10,000$, in the conservative option (Figure 3.3, fifth column) the graph was correctly estimated for the first three values os c (0.25, 0.5 and 1) and it was

underestimated for the other values. Now in the non conservative option (Figure 3.4, fifth column), just for $c = 2$ the graph was not correctly estimated, it was underestimated, for the other values of c it was correctly estimated.

Like in the first example, for Example 2, the non conservative approach was slightly better than the conservative one considering the sample sizes and values of c, having correctly estimated the graph 6 of 25 times against 4 of 25 times.

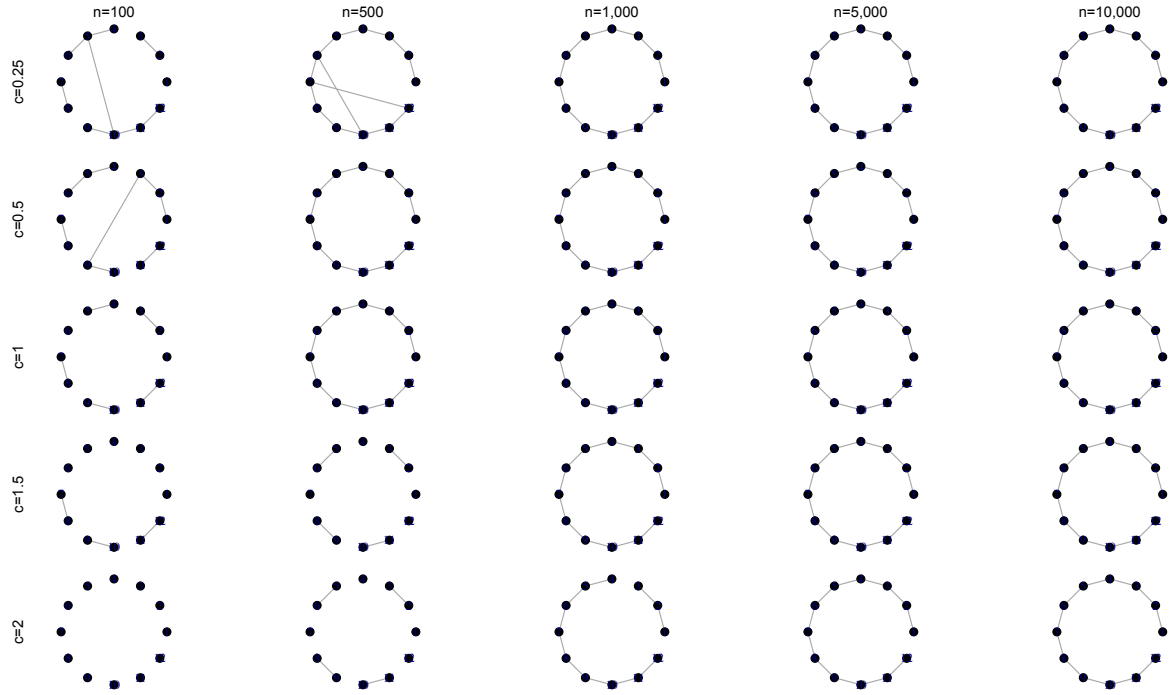

Figure 3.5: *Graph of example 3: Evolution of the estimated graph for different sample sizes and constant c considering the conservative option.*

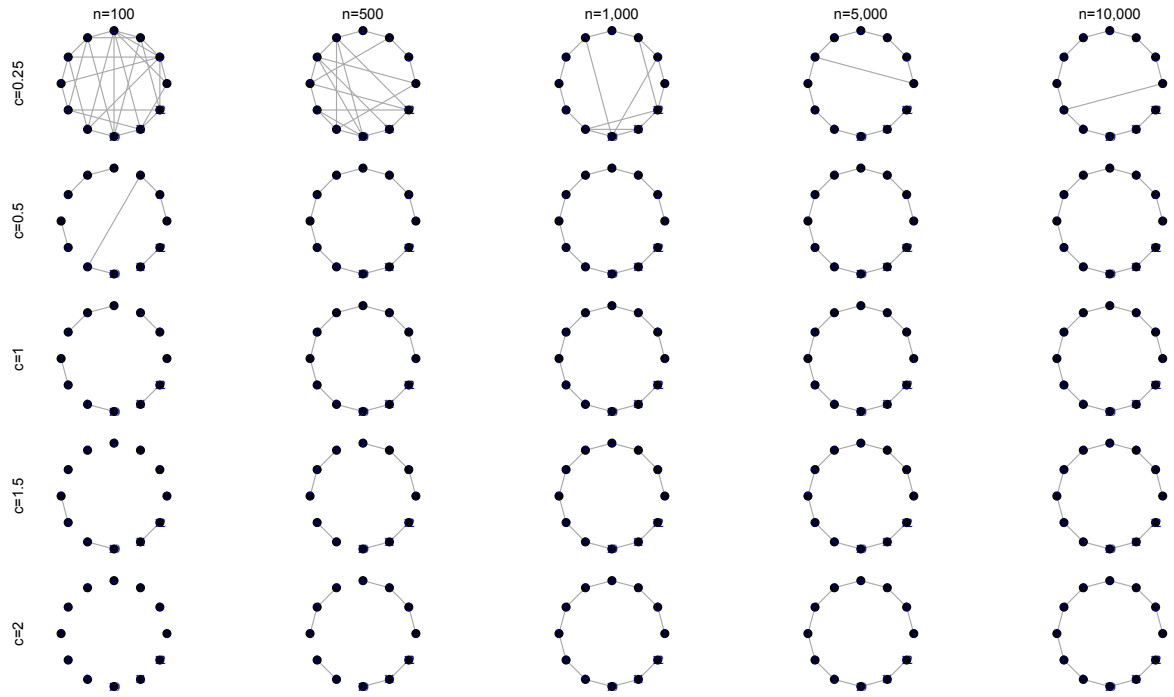

Figure 3.6: *Graph of example 3: Evolution of the estimated graph for different sample sizes and constant c considering the non conservative option.*

For example 3 the results are presented in Figures 3.5 and 3.6, the conservative and non conservative options respectively. Comparing the results with the correct dependence structure of the graph in Figure 2.3 (Chapter 2) it is possible to see that:

- For the sample with size $n = 100$ the conservative option (Figure 3.3, first column) and the the non conservative option (Figure 3.4, first column) were not capable of reconstructing the true graph structure. The biggest difference between the two options is for $c = 0.25$, note that for the conservative option there is only one spare edge and for the non conservative one there are 15 spare edges.

- For the sample with size $n = 500$, for the conservative option (Figure 3.3, second column) the graph was overestimated for $c = 0.25$, it was correctly estimated for $c = 0.5$ and 1, and it was underestimated for the other values of c. The same happened for the non conservative option (Figure 3.4, second column), but for $c = 0.25$ the number of spare edges was bigger (two against ten edges).

- For the sample with size $n = 1,000$, for the conservative option (Figure 3.3, third column) the graph was correctly estimated for $c = 0.25$, 0.5, 1 and 1.5, and it was underestimated, with one missing edge, for $c = 2$. For the non conservative option (Figure 3.4, third column) the graph was correctly estimated for $c = 0.5$, 1, 1.5 and 2, and it was overestimated, with six spare edges for $c = 0.25$.

- For the samples with sizes $n = 5,000$ and $n = 10,000$, for the conservative option (Figure 3.3, fourth and fifth columns) the graph was correctly estimated for all values of c. And for the non conservative option (Figure 3.4, fourth and fifth columns) the graph was correctly estimated for $c = 0.5$, 1, 1.5 and 2, and it was overestimated, with one spare edge, for $c = 0.25$.

Unlike what happened in the two previous examples, in this one the conservative option was slightly better than the non conservative one, having correctly estimated the graph 16 of 25 times against 14 of 25.

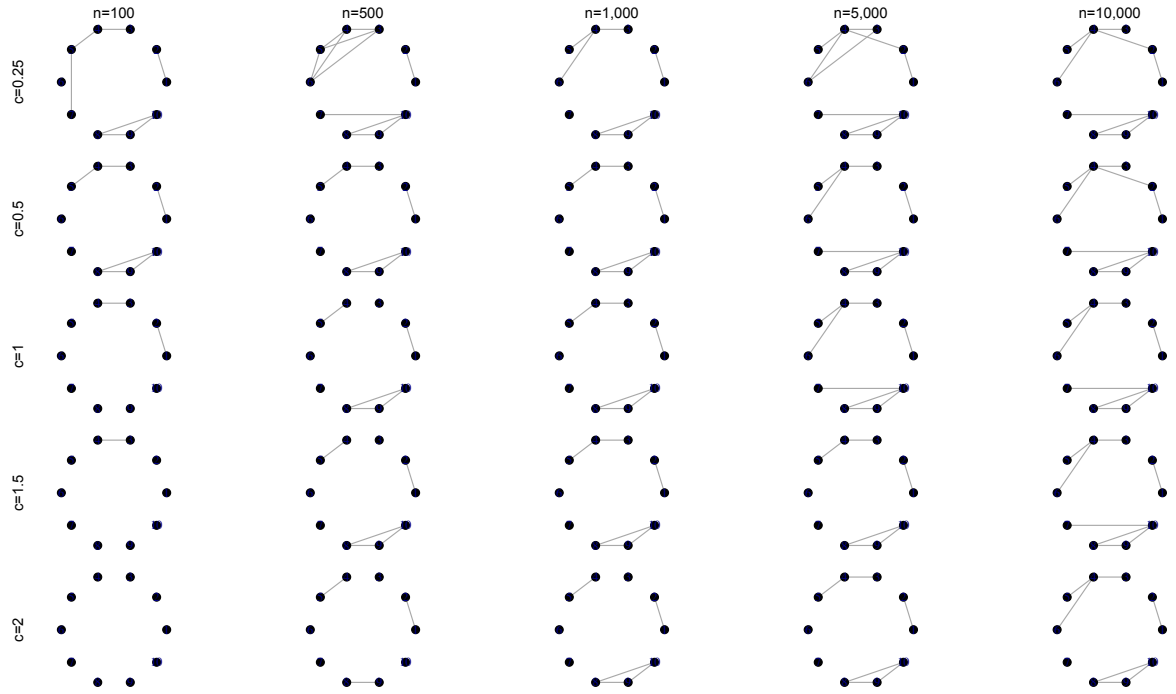

Figure 3.7: *Graph of example 4: Evolution of the estimated graph for different sample sizes and constant c considering the conservative option.*

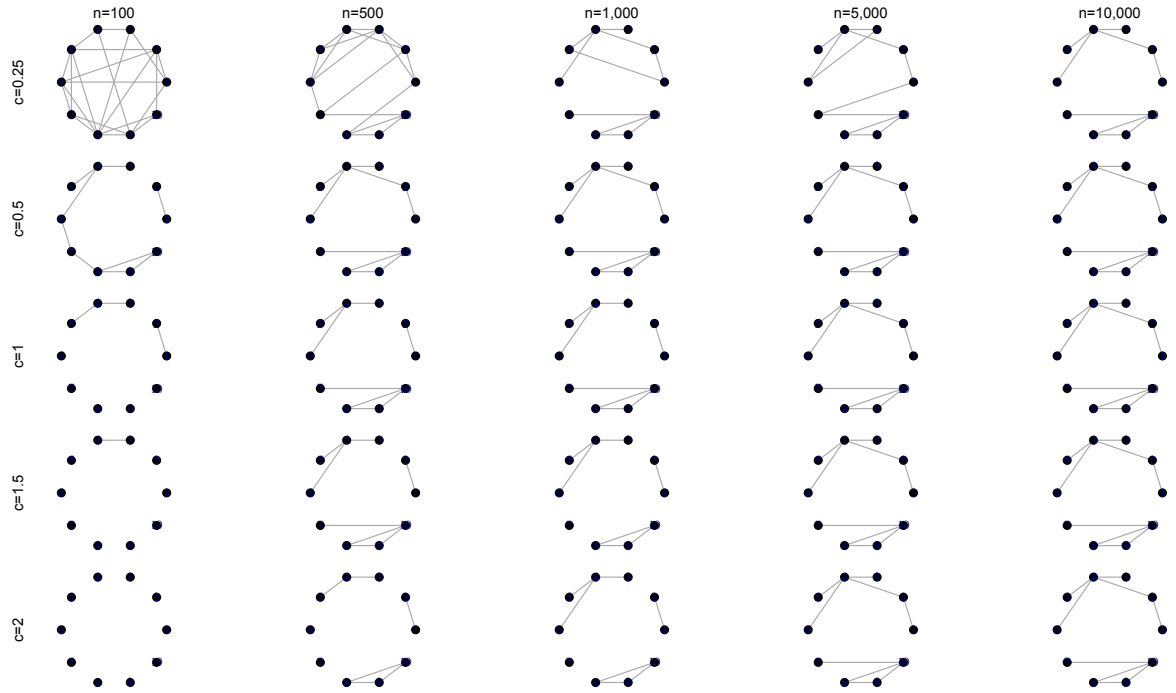

Figure 3.8: *Graph of example 4: Evolution of the estimated graph for different sample sizes and constant c considering the non conservative option.*

The results of the last example, Example 4, are presented in Figures 3.7, the conservative option, and Figure 3.8, the non conservative option. Compared to Figure 2.4 in Chapter 2, that contains the real structure of the graph, with the results presented here it is possible to observe that:

- For the sample with size $n = 100$, in the conservative option (Figure 3.1, first column) the graph was underestimated for all values of c. Note that for $c = 2$ there are no edges, this kind of graph means that all variables are independent. And in the non conservative option (Figure 3.1, first column) the graph was overestimated for $c = 0.25$, for $c = 0.5$ there is two missing edges and two spare edges, for the other values of c the graph was underestimated. As in the other option, here for $c = 2$ the graph has no edges. None of the options were able to estimate the correct graph structure.

- For the sample with size $n = 500$, in the conservative option (Figure 3.7, second column) for all values of c the graph was not correctly estimated, for $c = 0.25$ there are missing and spare edges, and for the other values of c the graph was underestimated. In the non conservative option (Figure 3.8, second column) the graph was correctly estimated only for $c = 0.5$, it was overestimated for $c = 0.25$ and underestimated for $c = 1.5$, 1 and 2.

- For the sample size $n = 1,000$, in the conservative option (Figure 3.7, third column) the graph was underestimated for all values of c. And in the non conservative option (Figure 3.8, third column), the graph was correctly estimated only for $c = 0.5$, it was overestimated for $c = 0.25$, with one spare edge, and it was underestimated for the other values of c.

- For the sample size $n = 5,000$, in the conservative option (Figure 3.7, fourth column) the graph was overestimated for $c = 0.25$ and it was underestimated for the other values of c. In the non conservative option (Figure 3.8, fourth column) the graph was correctly estimated for $c = 0.5$, 1, 1.5 and 2, and it was overestimated for $c = 0.25$.

- For the sample size $n = 10,000$, in the conservative option (Figure 3.7, fifth column) the graph was correctly estimated for $c = 0.25$ and 0.5, and it was underestimated for the other values of c. And in the non conservative option (Figure 3.8, fifth column) the graph was correctly estimated for all values of c.

For this example, as in Examples 1 and 2, the non conservative approach seems to be better than the conservative one, having correctly estimated the graph 11 of 25 times against 2 of 25 times.

Throughout the examples it is possible to notice the effect of the sample size, the choice of the estimator constant c and if the estimator must be conservative or not. In summary, the increase of the sample size improves the estimation of the graph; the higher the value of the constant more evidence is needed to perceive the existence of edges, i.e., the choice of the constant makes the estimator more or less conservative; and even the choice between a conservative estimator or not impacts on the perception of the edges, a conservative estimator needs more information to put an edge between two vertices.

3.3.2 Under and Overestimation errors

In this evaluation it was used a constant $c = 1$, and the sample size ranged from 100 to 10,000. To estimate the errors, for each sample size they were generated 20 samples, and then it was calculated the average of the errors. The understimation error (ue) is given by the number of missing edges over the number of edges that the graph has, the overestimation error (oe) is given by the number of spare edges over the number of edges that the graph does not have, and a total error (te) is given by the sum of missing edges and spare edges over the total number of edges of the complete graph. That is, let $\widehat{G} = (V, \widehat{E})$ be an estimative of the graph $G = (V, E)$, then:

$$ue = \frac{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \in E \text{ and } (v,w) \notin \widehat{E}\}}{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \in E\}},$$

$$oe = \frac{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \notin E \text{ and } (v,w) \in \widehat{E}\}}{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \notin E\}} \text{ and}$$

$$te = \frac{\sum_{v \in V} \sum_{v<w} \left(\mathbf{1}\{(v,w) \in E \text{ and } (v,w) \notin \widehat{E}\} + \mathbf{1}\{(v,w) \notin E \text{ and } (v,w) \in \widehat{E}\} \right)}{\frac{|V|(|V|-1)}{2}}.$$

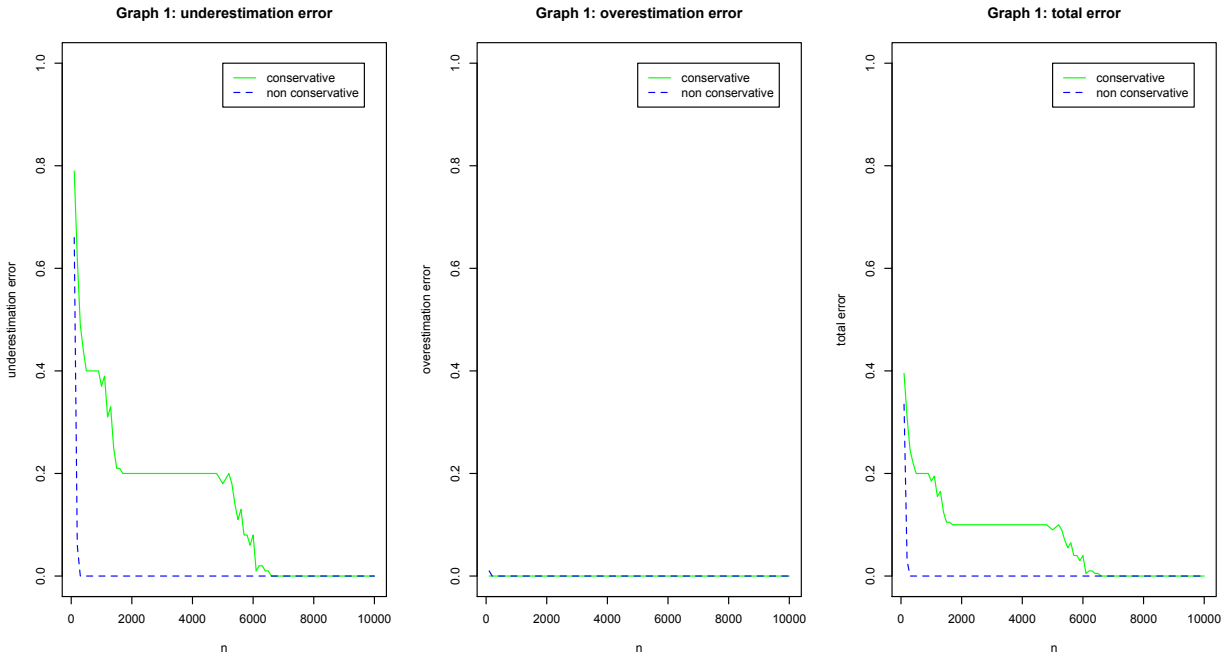

Figure 3.9: *Graph of example 1: Estimative of the underestimation, overestimantion and total errors for different sample sizes with $c = 1$.*

The comparison of the different types of errors was made for each one of the four examples we

are analyzing, and the results can be observed in Figures 3.9, 3.10, 3.11 and 3.12. And to calculate the errors the function `errors` in B.2 written in R language can be used.

For the first example, Figure 3.9, it is possible to see how the underestimation error of the non conservative (blue dashed line) choice decreases faster than the error of the conservative one (green continuous line). The overstimation errors were almost zero even for small sample sizes and in both cases (conservative and non conservative choices), and because of this, the total errors reflects only the behavior of the underestimation errors. So, for this example the non consevative seems to be the best choice.

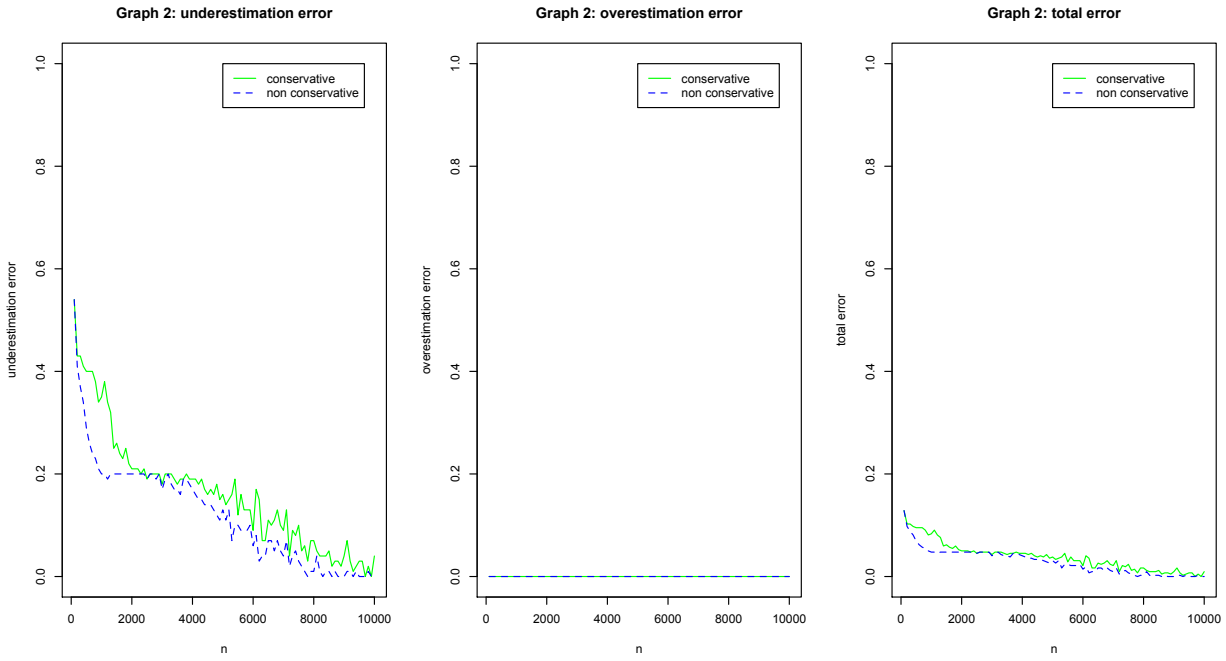

Figure 3.10: *Graph of example 2: Estimative of the underestimation, overestimantion and total errors for different sample sizes with* $c = 1$.

Looking at the graphics in Figure 3.10, the results for the second example, we can see again that even for small sample sizes the overestimation errors in both cases are zero, and despite the underestimation errors being closer in this example, the errors of the non conservative choice are smaller than the errors of the conservative one. So, again the non conservative choice seems to be the best choice.

For the third example, Figure 3.11, the understimation errors decrease really fast with the increasing sample size, but again the non conservative choice has a better performance than the conservative one. Because the overestimation errors were zero for both choices the total errors have the same behavior of the underestimation errors. In conclusion, the non conservative choice proved to be the best choice.

In the last example, Figure 3.12, we have the same results for the overestimation errors for both choices, the errors are zero, and looking at the underestimation errors the non conservative choice has a much better performance than the conservative one.

We should keep in mind that the errors are affected not only by the sample size but also by the choice of the constant c. It seems that the value 1 for the constant is large enough to prevent the overestimation of edges. Perhaps with a lower value ofthe constant c the overestimation errors could be observed, but this also would make the underestimation errors to be smaller.

3.3.3 ROC curves

The ROC (Receiver operating characteristic) curve was develop during World War II to detect electronic signals and problems with radars (Zweig and Campbell, 1993). Its goal was to quantify

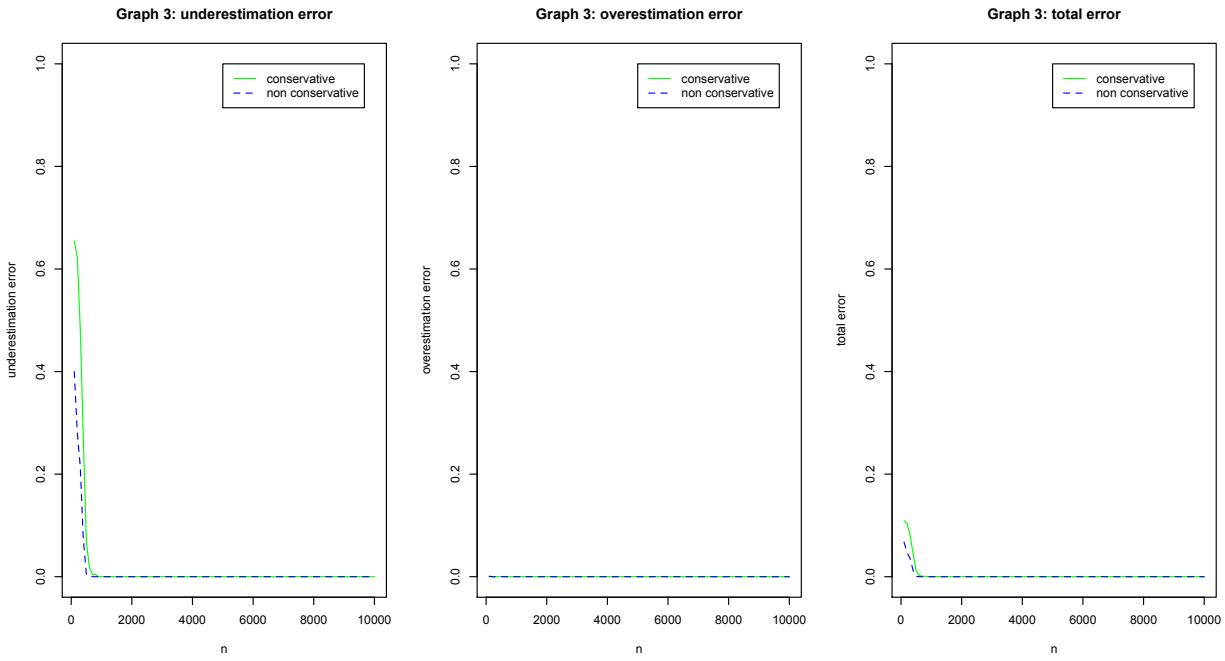

Figure 3.11: *Graph of example 3: Estimative of the underestimation, overestimantion and total errors for different sample sizes with* $c = 1$.

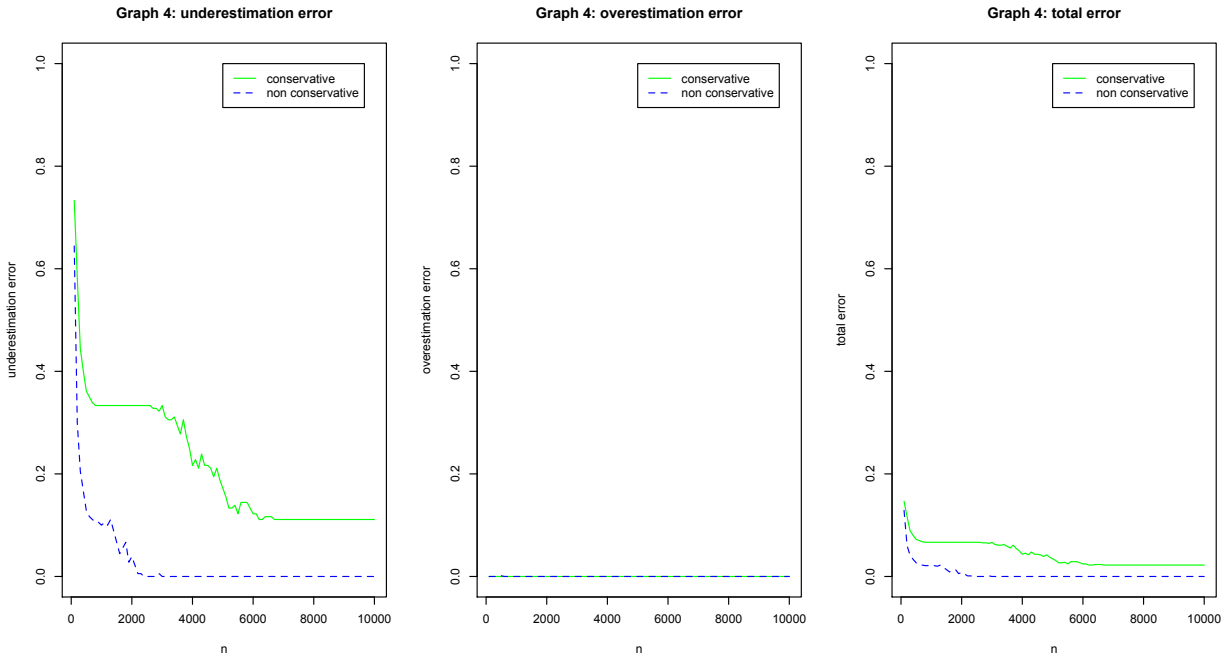

Figure 3.12: *Graph of example 4: Estimative of the underestimation, overestimantion and total errors for different sample sizes with* $c = 1$.

the ability of radar operators (originally called *receiver operators*) to distinguish a signal from a noise (Reiser and Faraggi, 1997). This ability was called *receiver operating characteristic* (ROC). When the radar detected something approaching, it was up to the operator to decide whether what was captured was, for example, an enemy aircraft (the signal), or some other irrelevant flying object, like a flock of birds (the noise) (Collinson, 1998). In the 1960's, ROC curves were used in experimental psychology and, in the 1970's, the methodology has spread widely in various fields of biomedical research, an area in which the goal became basically to assist the classification of

Graph	Neighborhood estimator		
	Identifies the edge	Does not identify the edge	
Has the edge	True positive	False negative	Number of existing edges
Does not have the edge	False positive	True negative	Number of non existing edges

Table 3.1: *Confusion matrix*

individuals in sick or not sick. In our context, the neighborhood estimator proposed here acts as a classifier that identifies the existence of edges between two vertices.

Translating the concepts to our case, to construct the ROC curve two definitions must be understood: the *sensibility*, defined as the probability of identifying an edge given that the edge exists, also called as true positive rate; and the *specificity*, defined as the probability of do not identifying an edge given that the edge does not exist, and 1− specificity is also known as false positive rate. The ROC curve of a classifier is the pair (1− specificity, sensibility).

The ideal is to have a curve above the line represented by $y = x$, i.e., a classifier that has sensibility greater than 1− specificity. And when comparing the curves of two or more classifiers, if one curve dominate the others the classifier represented by this dominant curve is considered better than the others. But when this dominance is not clear, the choose of a better classifier should consider other factors that may be theoretical and/or practical.

The confusion matrix, in Table 3.1, is a common representation of the results of a classification method which contains the successes and failures of a rating. Now, let $\widehat{G} = (V, \widehat{E})$ be an estimative of the graph $G = (V, E)$, then:

$$\text{Sensibility } (\widehat{G}) = \frac{\text{True positive}}{\text{Number of existing edges}} = \frac{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \in E \text{ and } (v,w) \in \widehat{E}\}}{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \in E\}} \text{ and}$$

$$1 - \text{Specificity } (\widehat{G}) = \frac{\text{False positive}}{\text{Number of non existing edges}} = \frac{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \notin E \text{ and } (v,w) \in \widehat{E}\}}{\sum_{v \in V} \sum_{v<w} \mathbf{1}\{(v,w) \notin E\}}.$$

This pair can be calculated using the function `roc.curve` in B.2 written in R language.

To compare the conservative and non conservative approaches of the estimator using the ROC curve, they were used different values of the constant c, in such a way that each point of the curve was estimated by the average of the pair (1− specificity, sensibility) given by 50 samples for a particular value of c. The sample size was kept the same ($n = 100$), so the only thing that varies is the constant c ranging from 0 to 3.5.

Figures 3.13, 3.15 and 3.16, containing the results of Examples 1, 3 and 4, respectively, clearly show the dominance of the non-conservative estimator (blue asterisks) over the conservative estimator (green dots) resulting in a better classifier. In Figure 3.14, containing the results of Example 2, this dominance is not clear. It seems that for a set of values of c the non conservative estimator is slightly better than the conservative one. But reminding that in previous evaluation the errors of the two types of estimators were close, the results observed here seem consistent with previous ones.

Taking into account all that was seen and discussed in this section about the performance of the estimator, it can be concluded that the best choice is the non conservative approach. This will be considered in the next chapter (Chapter 4), where the estimator is used in real data sets.

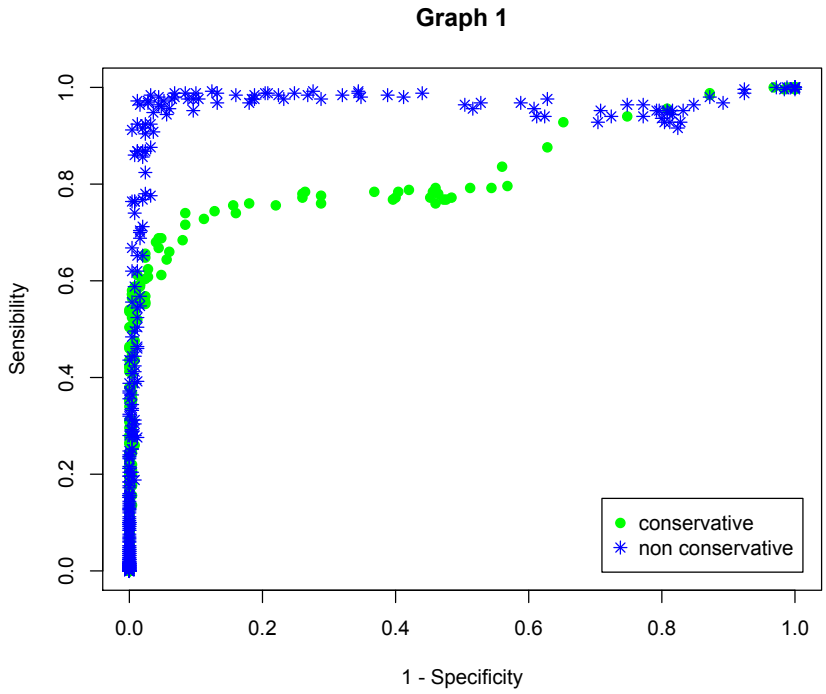

Figure 3.13: *Graph of example 1: ROC curves for* $cin[0;\ 3.5]$ *and* $n = 100$.

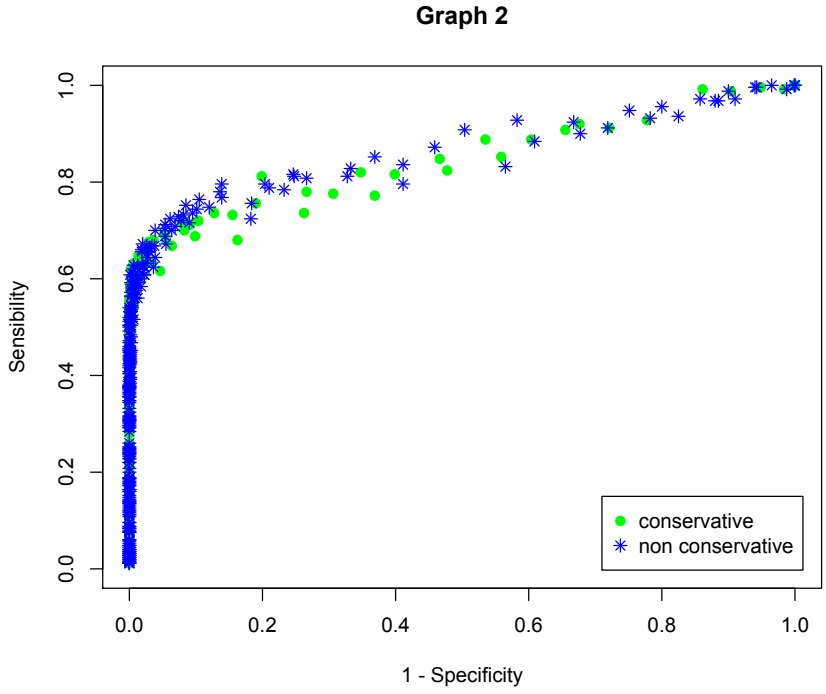

Figure 3.14: *Graph of example 2: ROC curves for* $cin[0;\ 3.5]$ *and* $n = 100$.

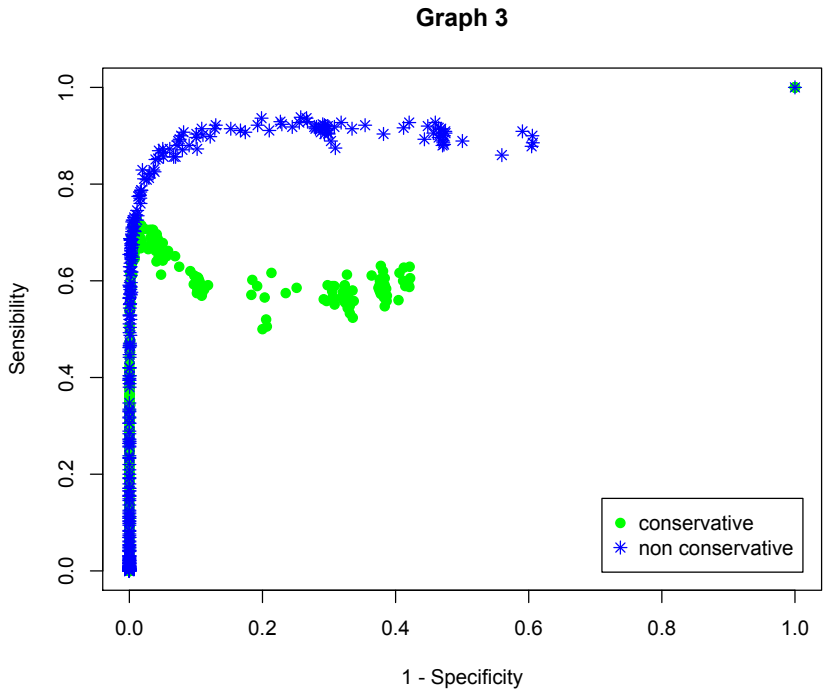

Figure 3.15: *Graph of example 3: ROC curves for* $cin[0;\ 3.5]$ *and* $n = 100$.

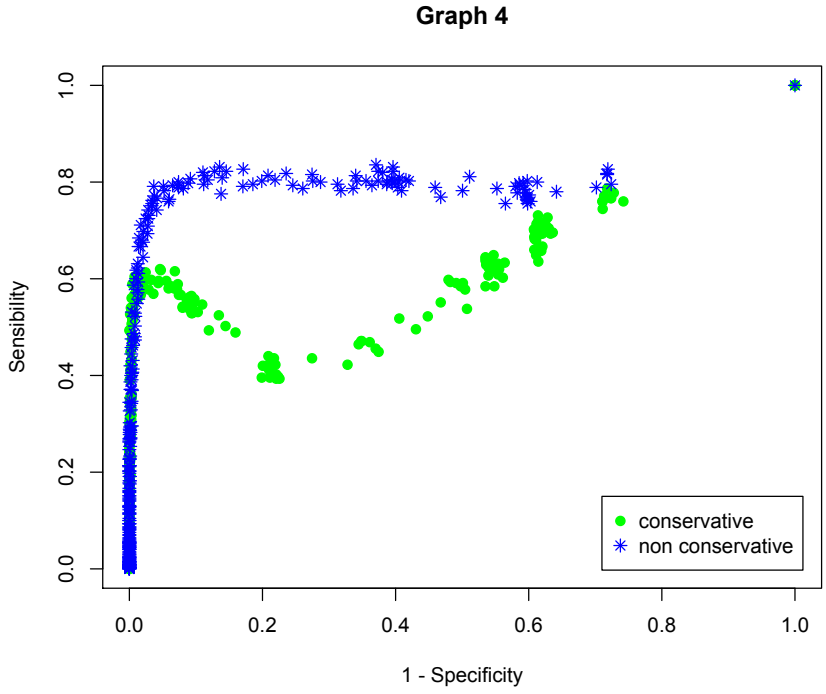

Figure 3.16: *Graph of example 4: ROC curves for* $cin[0;\ 3.5]$ *and* $n = 100$.

Chapter 4

Aplications

This Chapter presents two aplications of random Markov fields in real data sets. The first one is an aplication in stock market indexes of five countries, that tries to understand the dependency relationships of the rise or fall of these indexes. The second aplication uses EEG (electroencephalogram) signals, captured by 20 electrodes during a visual experiment, and tries to find out the dependency relationships of these signals from different regions of the brain.

As already mentioned, we used here the results obtained in the previous chapter (Chapter 3), that is, we used the non conservative approach for the estimator. However it is necessary to choose a value for the constant c, and to solve the question of what is the best value for the constant c we use a common resamplig method called cross-validation (see James et al. (2014) and Hastie et al. (2009)). The next section explain this method and how it is used in our context.

4.1 Cross-validation

There exists different approachs of this method, as the validation set approach, the leave-one-out cross-validation and the k-fold cross-validation. Here we use the k-fold cross-validation (James et al. (2014) and Hastie et al. (2009)). In this aproach the sample observations are randomly divided into k groups, or *folds*, of approximately equal size. In summary one of the k groups is treated as a validation set, and the remaining $k-1$ groups is the training set used to estimate the model. This procedure is repeated k times; each time, a different group of observations is treated as a validation set and the measure of interest is calculated. So, at the end, this process results in k estimates of this measure. And the k-fold cross-validation estimate is computed by averaging these values.

In linear models, for example, the measure of interest can be the mean squared error, but in our case we are interested in the log conditional likelihood of the model. So, for each value of c this measure is calculated by the discribed method, and then the best c is the one with the higher log conditional likelihood. Kepping in mind that here it is used the likelihood of the conditional distributions of each vertex $v \in V$ given its neighborhood, the log conditional likelihood of the model in the i-nth fold with constant c is:

$$\log \mathbb{P}(c)_i = \sum_{v \in V} \sum_{x_{\widehat{\mathrm{ne}}(v)} \in A^{\widehat{\mathrm{ne}}(v)}} \sum_{a \in A} N_i(a, x_{\widehat{\mathrm{ne}}(v)}) \log \hat{p}(a|x_{\widehat{\mathrm{ne}}(v)}) \,, \tag{4.1}$$

where the condinal probability $\hat{p}(a|x_{\widehat{\mathrm{ne}}(v)})$ and the neighborhood $\widehat{\mathrm{ne}}(v)$ are obtained in the $k-1$ groups used to train the model, and the number $N_i(a, x_{\widehat{\mathrm{ne}}(v)})$ comes from the i-nth group used to valitated the model (this can be calculated using the program in B.3). Then for each c, the measure that is used to compare and choose the best value of the constant is:

$$CV(c)_k = \frac{1}{k} \sum_{i=1}^{k} \log \mathbb{P}(c)_i \,. \tag{4.2}$$

Once the value $c_k^* = \max_c CV(c)_k$ is found, the graph is estimated using the constant c_k^* and the entire sample. For both aplications the samples were divided in $k = 10$ folds.

4.2 Stock market indexes

In this aplication we took the stock market indexes of five countries, to know: Germany, Australia, Brazil, Spain and Japan. More than 14 years of data[1] were collected, from April 18^{th} of 2001 to November 30^{th} of 2015. And, excluding weekends and national holidays, the total number of observations is 3.218 days. Figure 4.1 shows the variation of those indexes over the 3.218 days, and it is hard to see any possible correlation between the indexes of the countries. However, we can use a graph to represent the conditional dependency relationships between the ups and downs of these indexes.

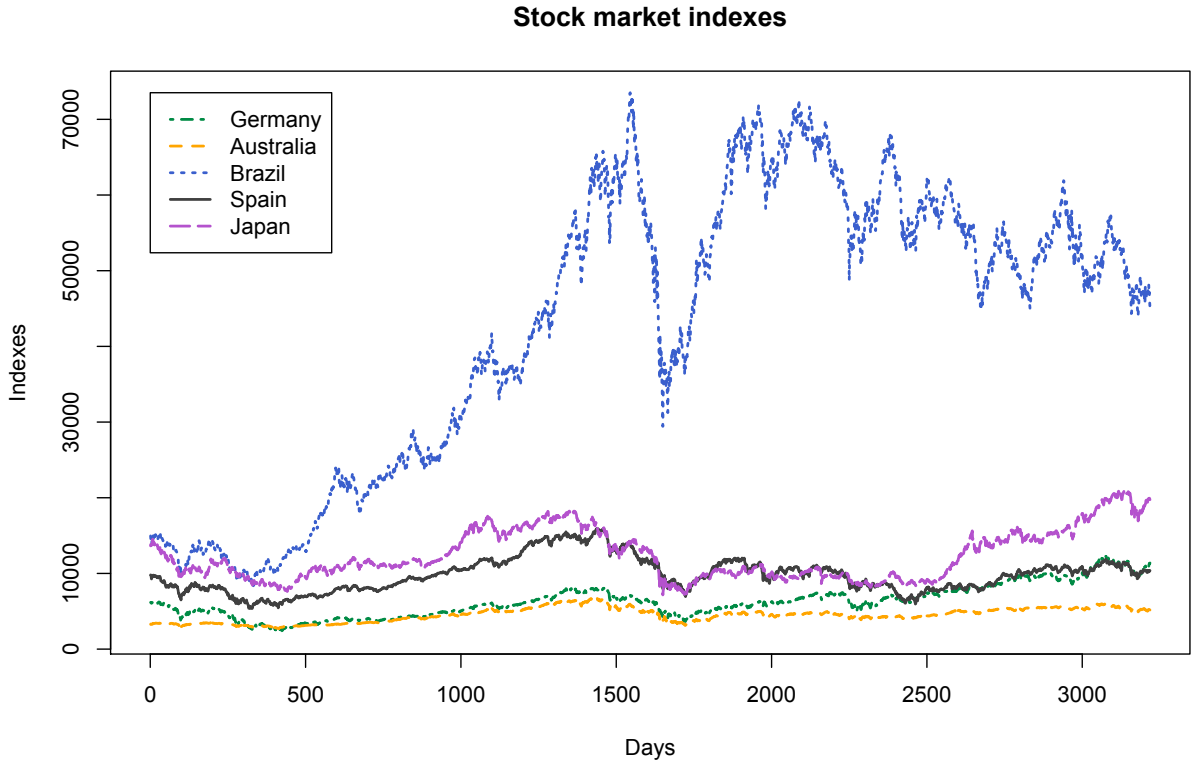

Figure 4.1: *Stock market indexes from the five countries, from April* 18^{th} *of 2001 to November* 30^{th} *of 2015.*

So, to use the model proposed in this thesis we have to transform the data of a continuous set to a finite alphabet, and we did as follow: if the index of day $d + 1$ is higher than the index of day d the new observation take the value 1, otherwise the new observation take the value 0. Then we have the alphabet $A = \{0, 1\}$ and 3.217 observations (one less than the total number of days, since we are evaluating the variation between days). One thing that we can not forget is that, even if we use this new observations that represents the variation between days, this observations can still not be independent, we will always have this kind of problems when dealing with time series data. To avoid this, some actions can be taken, as use only 10% of the observations, ensuring that the chosen observations have a gap of 9 observations between them. This gap can be found analyzing the autocorrelation of the observations, but even if this gap is less than 9, for example 5, we lose

[1] the data were obtained from this website http://br.investing.com/indices/major-indices

information, and sometimes we cannot afford this. For this aplication, even with this problem in mind, we decided to use the entire sample.

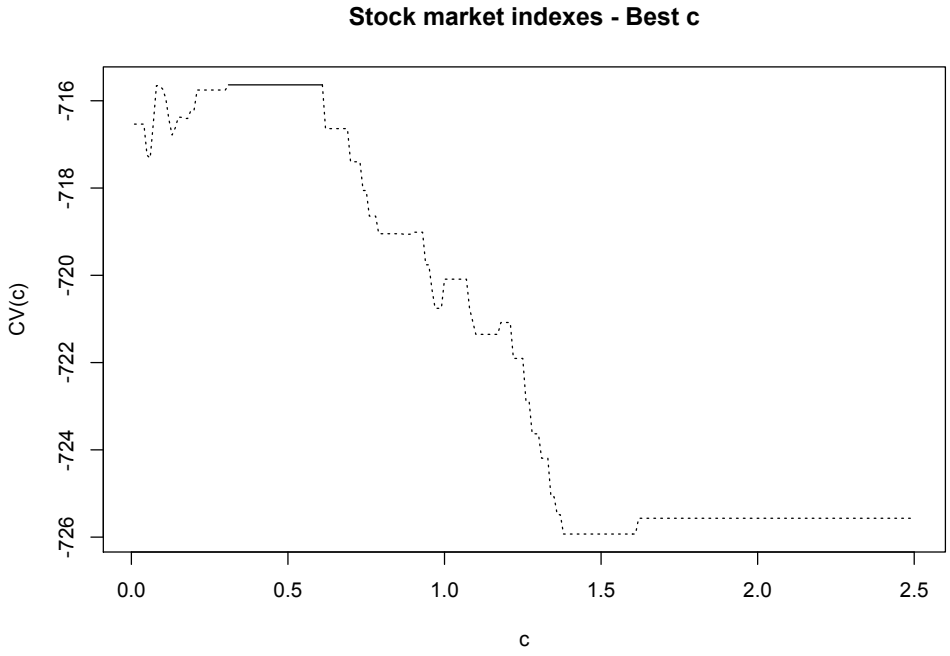

Figure 4.2: *Stock market indexes: $CV(c)_{10}$, for $c = [0.01, 2.5]$.*

Using the concepts of k-fold cross-validation explained in the previous section, and taking $k = 10$ as said, we performed the procedure to find the value c^* assessing 250 values for c, from 0.01 to 2.5, and Figure 4.2 shows the results. Note that the highest value obtained was $CV(c^*)_{10} = -715.6354$ for $c^* = [0.31, 0.61]$ (the continuous line).

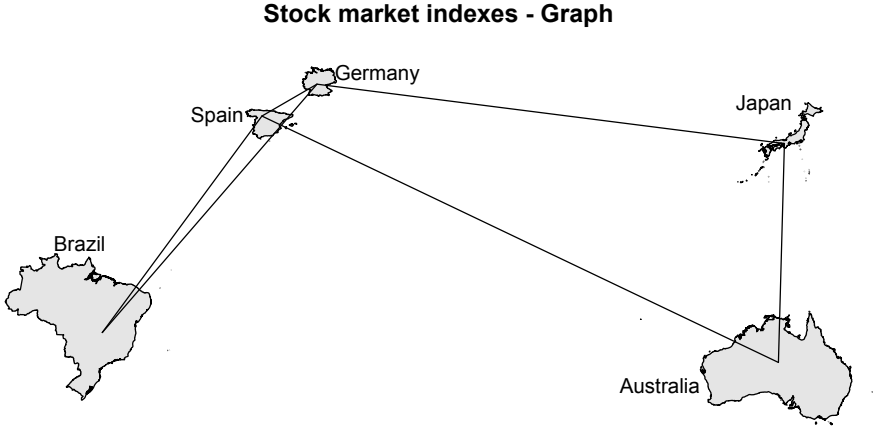

Figure 4.3: *Stock market indexes: graph that represents the relationshio between the countries stock market indexes.*

Estimating the graph considering the values of $c^* = [0.31, 0.61]$ and the whole sample we got the graph in Figure 4.3. This graph represents the relationship between the countries stock market indexes ups and downs. Note that Germany and Spain have three neighbors each, Germany has Brazil, Spain and Japan as its neighbors, and Spain has Brazil, Germany and Australia. And Brazil, Australia and Japan only have two neighbors each one, Brazil has Spain and Germany, Japan has Spain and Australia, and Australia has Spain and Japan. The result shows that the connections seem to be related to the geographical proximity of the countries.

4.3 EEG signals

An interesting application of the model is to study the relationships of dependence in neuroscience data. In this case, we study electroencephalograms (EEG) data, which are electric currents produced in the brain. The capture of these currents is performed through electrodes applied to the scalp, the brain surface, or even within the brain matter.

Here we use the data obtained by Prof. Claudia D. Vargas[2] group. These data were collected by 20 electrodes placed on the scalp, which are positioned using the international 10-20 system. By this system, the nomenclature of the electrodes is given according to the region where they are located: Fp = frontal polar, F = frontal, T = temporal, C = central, P = parietal and O = occipital. The electrodes placed on the line that goes from nose to the nape are indexed by the letter z (from zero), those located on the left side of this line are indexed by odd number and on the right side by even numbers. Figure 4.4 shows the positions of the electrodes and the nomenclature used in this study.

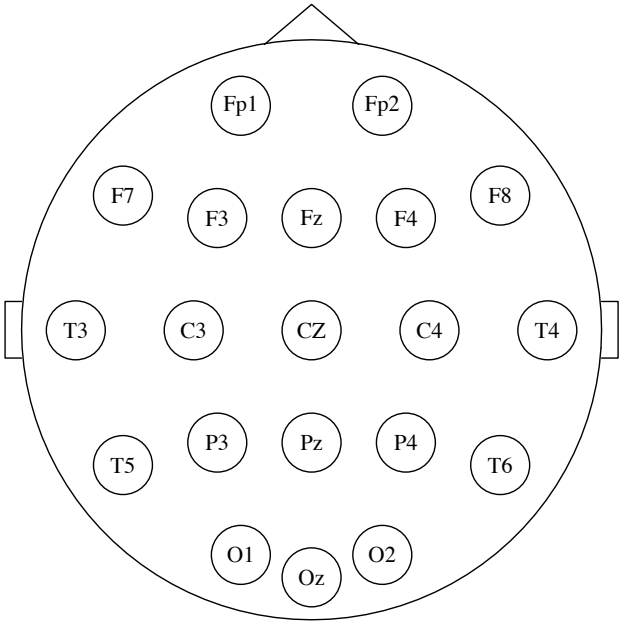

Figure 4.4: *EEG signals: Position of electrodes on the scalp.*

The data were collected during a visual experiment where the subject observes a short film in which a doll walks and then disappear, as it passed behind a wall. In these two moments, the walking and the occlusion, the EEG signals were captured. For the walking stage 1.000 points were recorded and for the occlusion stage 1.300 points were recorded in each one of the 20 electrodes. Altogether, 8 people participated in the experiment and each have different numbers of repetitions: 25, 41, 40, 38, 42, 47, 32 and 39 (a total of 304 repetitions).

The first question we have to solve is the choice of how to discretize the signals, as these are continuous. It is known that this type of signal shows brain activity in its peaks and valleys, so the

[2] from Federal University of Rio de Janeiro

choice of discretization took this information into account. Such a way that the peaks and valleys are marked with the value 1, while the other points were marked with 0. Then, to recognize these points the following procedure was done:

- for the signal of each combination stage (walking, occlusion), repetition $(1, 2, \ldots, 304)$ and eletrode $(1, 2, \ldots, 20)$, the first and third quartiles were calculated;
- if the point is less than the first quartil or if it is greater than the third quartil, the observation received the value 1, otherwise the observation received the value 0.

Doing that we transformed the continuos observations into an alphabet $A = \{0, 1\}$.

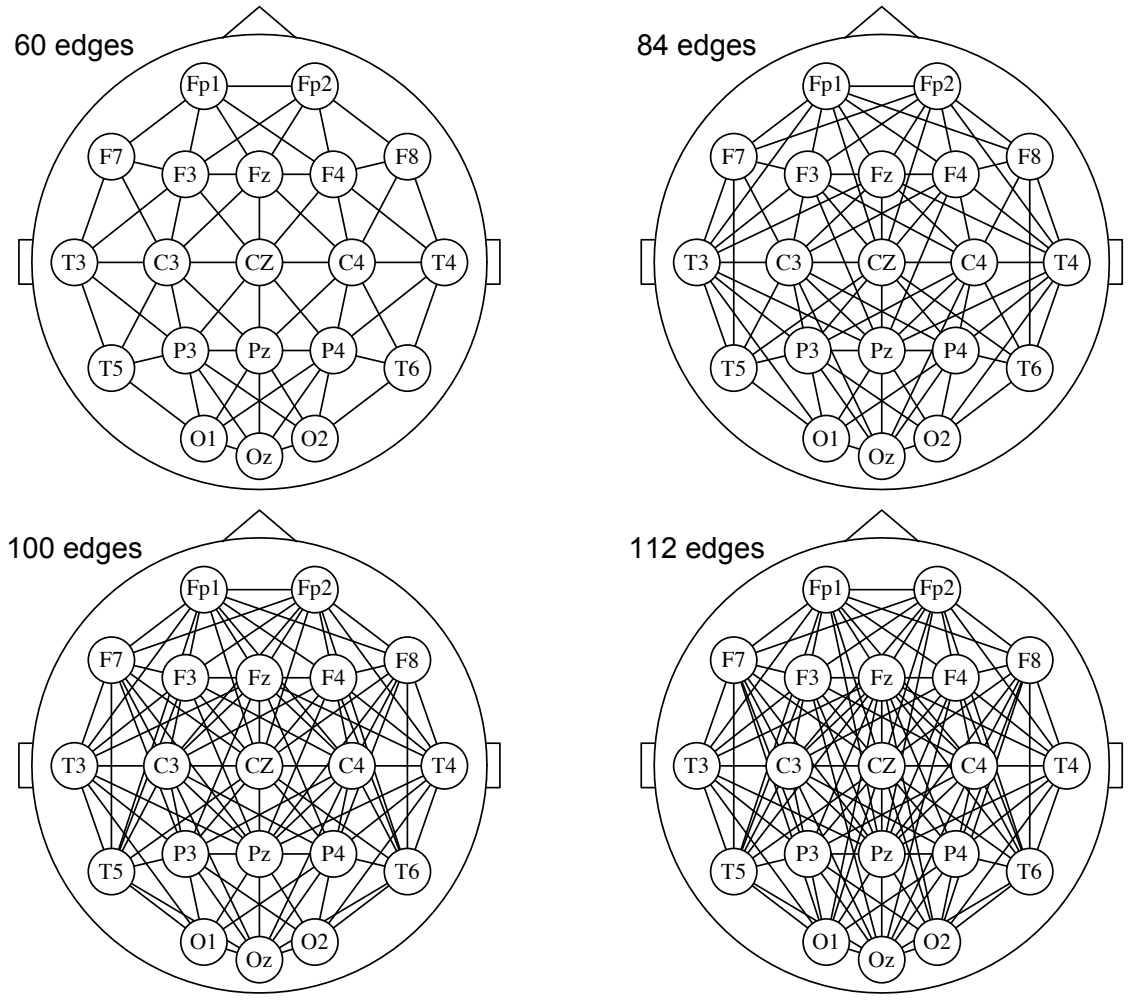

Figure 4.5: *EEG signals: The four different graphs used to start the model estimation.*

The second question to addrees is the lack of independence between observations. It is clear that within the sequence formed by these points there are no independence, and a preliminary assessment showed that the autocorrelation in these sequences are far from zero even for high lag values, such as 20 or 30. So to solve this problem we did the following: for each repetition of each stage we sellected from the original sample one observation every 100, ensuring a distance of 100 points between observations. Thereby, each repetition of the walking stage has size 10 and each repetition of the occlusion stage has size 13. Then we put together the elements of each repetition of each person into a single sample for each stage, so the sample sizes are 3,040 for the walking stage and 3,952 for the occlusion stage. Because the goal is to find for each stage the graph that represents the interactions between the signals captured by the electrodes and identify whether there are differences between the graphs.

As explain in Chapter 3, it is possible to choose a set of neighbors smaller than the set that represents the complet graph to be tested in the estimation program. Because of the large number of possible neighbors, 19 for each electrode, we decided to use the option of reduce the number of neighbors tested. We tested four different graphs to start the estimation, in each new test we increased the number of edges: starting with 60 edges, then 84, 100 and 112 edges (keep in mind that the complete graph has 190 edges). Figure 4.5 shows this four graphs.

Again we used the concept of k-fold cross-validation (with $k = 10$) to find the value c^* assesing 21 values, from 0.001 to 1. For each one of the four different inicial graphs and stages (walking and occlusion) the results for c^* was the same, we got $c^* = 0.001$. Estimating the models again using the entire samples and $c = 0.001$, the graphs estimated were equal to the graph used to start the estimation for both stages, no matter which graph was used to start the estimation.

We conclude that this model could not identify differences between the stages, if such differences exist. Note that other approaches to the treatment of samples or initiating the estimation by the complete graph could cause other results than those observed here.

Chapter 5

Generalization to Markov random fields with countable infinite set of vertices

The theoretical results presented until now were just intended for discrete Markov random fields over A^V with graph G, where V is a finite set. In this chapter we generalize these results to the case of a countable infinite set of variables.

5.1 Basic lemmas

Note that Lemmas 3 and 4 of Chapter 2 can not be used and must be changed to fit the new condition imposed by V countable infinite. So, despite the definition of a Markov random field being the same, the proof of the existence of a minimum Markov neighborhood can not be achived using the finite intersection of all Markov neighborhoods, but it is possible to use a given Markov neighborhood to define it. That is, for $v \in V$, let W_v be a given Markov neighborhood of v (that it is assumed to exist) and define

$$\Theta(v) = \{W \subset W_v : W \text{ is a Markov neighborhood of } v\}$$

Now, define as in Chapter 2,

$$\mathrm{ne}(v) = \bigcap_{W \in \Theta(v)} W. \tag{5.1}$$

Lemma 10. *For all $v \in V$, $ne(v)$ defined by (5.1) is a Markov neighborhood of v and $ne(v) \subset \Phi$ for all $\Phi \subset V$ Markov neighborhood of v.*

Proof. Because $\mathrm{ne}(v)$ is a finite intersection of Markov neighborhoods, by the Markov intersection property, $\mathrm{ne}(v)$ is a Markov neighborhood as it is the intersection of any Markov neighborhood Φ and W_v.

So, because $\Phi \cap W_v$ is a Markov neighborhood and $\Phi \cap W_v \subset W_v$, then $\mathrm{ne}(v) \subset \Phi \cap W_v$. On the other hand, $\Phi \cap W_v \subset \Phi$ wich implies that $\mathrm{ne}(v) \subset \Phi$, for all $\Phi \subset V$ Markov neighborhood of v. □

Recalling that the conditional distribution of X_v given $X_{\mathrm{ne}(v)} = a_{\mathrm{ne}(v)}$ is denoted by $\{p(a|a_{\mathrm{ne}(v)})\}_{a \in A}$; i.e., for all $a \in A$ we have

$$p(a|a_{\mathrm{ne}(v)}) = \mathbb{P}(X_v = a | X_k = a_k,\ k \in \mathrm{ne}(v))\,. \tag{5.2}$$

And similarly, the joint distribution of $(X_v, X_{\mathrm{ne}(v)})$ is denoted by $\{p(a_v, a_{\mathrm{ne}(v)})\}_{a_v \in A, a_{\mathrm{ne}(v)} \in A^{\mathrm{ne}(v)}}$. Here we present the generalization of Lemma 4 for the case of V countable infinite, as presented in Lemma A.2 of Csiszár and Talata (2006).

Lemma 11. *For a Markov random field over A^V, let $ne(v)$ be the smallest Markov neighborhood of $v \in V$. If a neighborhood W satisfies*

$$p(a|a_W) = p(a|a_{ne(v)})$$

then W is a Markov neighborhood.

Proof. We have to show that for any $\Delta \supset W$ finite with $\Delta \subset V$,

$$p(a|x_\Delta) = p(a|x_W). \tag{5.3}$$

As $\mathrm{ne}(v)$ is a Markov neighborhood, the lemma's condition implies

$$p(a|a_W) = p(a|a_{\mathrm{ne}(v)}) = p(a|a_{\mathrm{ne}(v)\cup\Delta}).$$

So (5.3) follows, because $W \subseteq \Delta \subseteq \mathrm{ne}(v) \cup \Delta$. □

The following section presents all the arguments, assumptions and definitions needed to handle the nuances of working with an countable infinite graph.

5.2 Model selection when V is countable infinite

Previously, a sample with size n of $\{X_v : v \in V\}$ meant, in simple words, a matrix of dimensions $n \times |V|$. Now for $i = 1, \ldots, n$ we assume the observation of an independent sample of $\{X_v : v \in V_n\}$, where V_n is a finite subset of V, $V_n \uparrow V$ when $n \to \infty$, i.e. $V_1 \subseteq V_2 \subseteq V_3 \subseteq \ldots$.

The estimation of the conditional probabilities for the sample stays the same, because we focus on finite subsets of V. The diferences happen in the estimation of the Markov neighborhoods and by consequence in the reconstruction of the graph. Actually now we just can reconstruct a finite subgraph of G, with a finite sample, because G has a countable infinite number of nodes.

Definition 12. *Given a function $c = c(n) > 0$, the empirical neighborhood of the vertex v is the set of indices $\widehat{ne}(v)$ given by*

$$\widehat{ne}(v) \;=\; \underset{W \subset V_n \setminus \{v\}}{\arg\max} \left\{ \log \hat{\mathbb{P}}_{v|W}(x_v^{(1:n)} | x_W^{(1:n)}) - c\,|A|^{|W|} \log_{|A|} n \right\}. \tag{5.4}$$

Definition 13. *Define $p_{min}(v)$ as*

$$p_{\min}(v) \;=\; \inf\{\, p(a|a_W) : p(a|a_W) > 0,\, a \in A,\, a_W \in A^W,\, W \subset V \text{ finite and } v \notin W\}. \tag{5.5}$$

Now consider the following assumptions:

Assumption 14. $\inf_{v \in V}\{p_{min}(v)\} = p_* > 0$.

Assumption 15. *The sequence $\{V_n : n \geq 1\}$ satisfies $|V_n| \leq o(\log n)$.*

Assumption 16. *For all $n \in \mathbb{N}$, $c > \frac{32|A|}{p_*^{|V_n|+1}}$.*

Assumption 17. *For all $v \in V$,*

$$\inf_{W \not\subset ne(v),\, ne(v) \not\subset W} D(p(\cdot|a_{ne(v)})\,;\, p(\cdot|a_W)) \geq \alpha > 0.$$

We prove the following consistency result for the neighborhood estimator.

Theorem 18. *Under the Assumptions 14, 15, 16 and 17, the estimator given by (5.4) satisfies $\widehat{ne}(v) = ne(v)$ eventually almost surely as $n \to \infty$.*

Here we are interested in estimating a finite subgraph of G. We still can estimate the neighborhood of each node and reconstruct the subgraph based on the set of neighborhoods. Given a set $V' \subset V$, we denote by $G_{V'}$ the induced subgraph; that is, the graph given by the pair (V', E'), where $E' = \{(v, w) \in E \colon v, w \in V'\}$. Based on the neighborhood estimator (5.4), we can construct an estimator of the subgraph $G_{V'}$ for any $V' \subset V$ finite, by defining the set of edges

$$\widehat{E}^{\wedge}_{V'} = \{(v, w) \in V' \times V' \colon v \in \widehat{\mathrm{ne}}(w) \text{ and } w \in \widehat{\mathrm{ne}}(v)\}.$$

The estimated subgraph will be the pair $\widehat{G}^{\wedge}_{V'} = (V', \widehat{E}^{\wedge}_{V'})$. In the same way, if we want to be less conservative we can define

$$\widehat{E}^{\vee}_{V'} = \{(v, w) \in V' \times V' \colon v \in \widehat{\mathrm{ne}}(w) \text{ or } w \in \widehat{\mathrm{ne}}(v)\}$$

and the estimated subgraph will be the pair $\widehat{G}^{\vee}_{V'} = (V', \widehat{E}^{\vee}_{V'})$.

Corollary 19. *For any finite set $V' \subset V$ we have $\widehat{G}^{\wedge}_{V'} = \widehat{G}^{\vee}_{V'} = G_{V'}$ eventually almost surely as $n \to \infty$.*

Now we prove the following theorem inspired in Theorema 1b) of Csiszár (2002) and we derived a corollary.

Theorem 20. *Let $V_n \subset V$, with $|V_n| \leq o(\log n)$. Then for all $\delta > 2$ eventually almost surely as $n \to \infty$*

$$\left| \frac{\hat{p}(a_W)}{p(a_W)} - 1 \right| \leq \sqrt{\frac{\delta \log n}{np(a_W)^2}}$$

simultaneously for all $W \subset V_n$ and $a_W \in A^W$.

Proof. For $a_W \in A^W$, let

$$Y_i = \mathbf{1}\{x^{(i)}_W = (a_W)\} - p(a_W) \; i = 1, 2, \ldots, n.$$

Note that $\mathbb{E}(Y_i) = 0$ and $Y_i \in [-1, 1]$, $\forall i = 1, 2, \ldots, n$, then by Hoeffding's Inequality (Appendix A, Theorem 28)

$$\mathbb{P}\left(\left| \frac{1}{n} \sum_{i=1}^{n} Y_i - \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} Y_i \right] \right| \geq t \right) \leq 2 \exp\left(- \frac{2n^2 t^2}{\sum_{i=1}^{n} (1 - (-1))^2} \right).$$

We have that

$$\frac{1}{n} \sum_{i=1}^{n} Y_i = \frac{N(a_W)}{n} - p(a_W)$$

and because $\mathbb{E}(Y_i) < \infty$,

$$\mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} Y_i \right] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}\left(Y_i\right) = \frac{1}{n} \sum_{i=1}^{n} 0 = 0.$$

Therefore,

$$\begin{aligned} &\mathbb{P}\left(\left| \frac{N(a_W)}{n} - p(a_W) \right| \geq t \right) \leq 2 \exp\left(- \frac{2n^2 t^2}{n4} \right) \\ &= 2 \exp\left(- \frac{nt^2}{2} \right) \end{aligned}$$

Take $t = \sqrt{\frac{\delta \log n}{n}}$, then

$$\begin{aligned}
&\mathbb{P}\left(\left|\frac{\hat{p}(a_W)}{p(a_W)} - 1\right| \geq \frac{t}{p(a_W)}\right) \\
&= \mathbb{P}\left(\left|\frac{\hat{p}(a_W)}{p(a_W)} - 1\right| \geq \sqrt{\frac{\delta \log n}{np(a_W)^2}}\right) \leq 2\exp\left(-\frac{\delta \log n}{2}\right)
\end{aligned}$$

Thus, we obtain

$$\begin{aligned}
&\mathbb{P}\left\{\left|\frac{\hat{p}(a_W)}{p(a_W)} - 1\right| \geq \sqrt{\frac{\delta \log n}{np(a_W)^2}} \text{ for some } W \subset V_n \text{ and } a_W \in A^W\right\} \\
&\leq \sum_{W \subset V_n} \sum_{a_W \in A^W} \mathbb{P}\left\{\left|\frac{\hat{p}(a_W)}{p(a_W)} - 1\right| \geq \sqrt{\frac{\delta \log n}{np(a_W)^2}}\right\} \\
&\leq 2^{|V_n|}|A|^{|V_n|} 2\exp\left(-\frac{\delta \log n}{2}\right) \\
&\leq 2^{o(\log n)}|A|^{o(\log n)} 2\exp\left(-\frac{\delta \log n}{2}\right)
\end{aligned}$$

which is summable in n for $\delta > 2$. This completes the proof. □

Corollary 21. *Suppose Assumptions 14 and 15 hold. Then for all $\delta > 16$, eventually almost surely as $n \to \infty$ we have*

$$|\hat{p}(a|a_W) - p(a|a_W)| \leq \sqrt{\frac{\delta \log n}{p_*^{|W|+2} N(a_W)}}$$

simultaneously for all $v \in V$, $W \subset V_n \setminus \{v\}$, $a \in A$ and $a_W \in A^W$.

Proof. Observe that

$$\left|\hat{p}(a|a_W) - p(a|a_W)\right| = \left|\frac{N(a, a_W)}{N(a_W)} - \frac{p(a, a_W)}{p(a_W)}\right|.$$

By summing and subtracting $N(a, a_W)/np(a_W)$ we obtain

$$\begin{aligned}
\left|\hat{p}(a|a_W) - p(a|a_W)\right| &\leq \left|\frac{N(a, a_W)}{N(a_W)} - \frac{N(a, a_W)}{np(a_W)}\right| + \left|\frac{N(a, a_W)}{np(a_W)} - \frac{p(a, a_W)}{p(a_W)}\right| \\
&= \frac{N(a, a_W)}{N(a_W)}\left|1 - \frac{N(a_W)}{np(a_W)}\right| + p(a|a_W)\left|\frac{N(a, a_W)}{np(a, a_W)} - 1\right| \\
&\leq \left|1 - \frac{N(a_W)}{np(a_W)}\right| + \left|\frac{N(a, a_W)}{np(a, a_W)} - 1\right| \\
&= \left|1 - \frac{\hat{p}(a_W)}{p(a_W)}\right| + \left|\frac{\hat{p}(a, a_W)}{p(a, a_W)} - 1\right|
\end{aligned}$$

Therefore, by Theorem 20 we have that for all $\delta > 2$, eventually almost surely as $n \to \infty$ we have

$$\begin{aligned}
\left|\hat{p}(a|a_W) - p(a|a_W)\right| &\leq \sqrt{\frac{\delta \log n}{np(a_W)^2}} + \sqrt{\frac{\delta \log n}{np(a, a_W)^2}} \\
&\leq \sqrt{\frac{4\delta \log n}{np(a, a_W)^2}}
\end{aligned}$$

simultaneously for all $v \in V$, $W \subset V_n \setminus \{v\}$, $a \in A$ and $a_W \in A^W$. But Theorem 20 and Assumption 14 also implies that eventually almost surely as $n \to \infty$,

$$np(a, a_W) = np(a|a_W)p(a_W) \;\geq\; p(a|a_W)N(a_W)/2$$

for all $v \in V$, $W \subset V_n \setminus \{v\}$, $a \in A$ and $a_W \in A^W$. Then, as $p(a|a_W)p(a, a_W) \geq p_*^{|W|+2}$ we obtain

$$\begin{aligned} \left|\hat{p}(a|a_W) - p(a|a_W)\right| \;&\leq\; \sqrt{\frac{8\delta \log n}{p(a|a_W)p(a, a_W)N(a_W)}} \\ &\leq\; \sqrt{\frac{8\delta \log n}{p_*^{|W|+2}N(a_W)}} \end{aligned}$$

and the statement of the corollary follows.

□

Proof of Theorem 18. Denote by

$$\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;=\; \log \hat{\mathbb{P}}_{v|W}(x_v^{(1:n)}|x_W^{(1:n)}) - c|A|^{|W|} \log n\,.$$

where

$$\hat{\mathbb{P}}(x_v^{(1:n)}|x_W^{(1:n)}) = \prod_{a_W \in A^W} \prod_{a \in A} \hat{p}(a|a_W)^{N(a,a_W)}\,.$$

If $V_n \setminus \{v\}$ is the bounding set for the neighborhood of vertex v and $\mathrm{ne}(v)$ is the minimal neighborhood of v, we have to prove that

$$\max_{W \subset V_n \setminus \{v\}, W \neq \mathrm{ne}(v)} \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely $n \to \infty$. And again, we divide the proof in three cases, showing that

$$\max_{W \in \mathcal{B}_i} \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

where

(a) $\mathcal{B}_1 = \{W \subset V_n \setminus \{v\} \colon \mathrm{ne}(v) \subset W\}$

(b) $\mathcal{B}_2 = \{W \subset V_n \setminus \{v\} \colon W \subset \mathrm{ne}(v)\}$

(c) $\mathcal{B}_3 = \{W \subset V_n \setminus \{v\} \colon W \not\subset \mathrm{ne}(v),\, \mathrm{ne}(v) \not\subset W\}$.

Compared to the proof of Theorem 7, the biggest changes in this proof are in cases (a) and (c), where the cardinality of the sets $\mathcal{B}$ is unbounded. In case (b) the only changes are the substitutions of Lemmas 3 and 4 by Lemmas 10 and 11, respectively. Next is the proof of each case:
(a). We have to prove that

$$\max_{W \in \mathcal{B}_1} \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely $n \to \infty$.

Observe that for all $W \in \mathcal{B}_1$

$$\begin{aligned} \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) - \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;&= \\ c\left(|A|^{|W|} - |A|^{|\mathrm{ne}(v)|}\right) \log n \;&- \sum_{a, a_W \in A^{|W|+1}} N(a, a_W) \log \frac{\hat{p}(a|a_W)}{\hat{p}(a|a_{\mathrm{ne}(v)})}\,. \end{aligned} \tag{5.6}$$

As these empirical probabilities are the maximum likelihood estimators we have that

$$\begin{aligned}\sum_{a,a_W\in A^{W+1}} N(a,a_W)\log\hat{p}(a|a_{\mathrm{ne}(v)}) &\geq \sum_{a,a_W\in A^{W+1}} N(a,a_W)\log p(a|a_{\mathrm{ne}(v)})\\ &= \sum_{a,a_W\in A^{W+1}} N(a,a_W)\log p(a|a_W)\,.\end{aligned}$$

Therefore, (5.6) can be lower-bouded by

$$c\left(1-\frac{1}{|A|}\right)|A|^{|W|}\log n \;-\sum_{a,a_W\in A^{W+1}} N(a,a_W)\log\frac{\hat{p}(a|a_W)}{p(a|a_W)}\,.$$

Now, observe that

$$\sum_{a,a_W\in A^{W+1}} N(a,a_W)\log\frac{\hat{p}(a|a_W)}{p(a|a_W)} = \sum_{a_W\in A^W} N(a_W)D(\hat{p}(\cdot|a_W)\,;\,p(\cdot|a_W))\,,$$

where D denotes the *Küllback-Leibler divergence* (see Definition 22 in Appendix A). Therefore we have, by Lemma 24 (in Appendix A)

$$\begin{aligned}&\sum_{a_W\in A^W} N(a_W)D(\hat{p}(\cdot|a_W)\,;\,p(\cdot|a_W))\\ &\qquad\leq \sum_{a_W\in A^W} N(a_W)\sum_{a\in A}\frac{[\,\hat{p}(a|a_W)-p(a|a_W)\,]^2}{p(a|a_W)}.\end{aligned}$$

Then, by Corollary 21 we have for any $\delta>16$, that

$$\begin{aligned}&\sum_{a_W\in A^W} N(a_W)D(\hat{p}(\cdot|a_W)\,;\,p(\cdot|a_W))\\ &\qquad\leq \sum_{a_W\in A^W} N(a_W)\sum_{a\in A}\frac{\delta\log n}{N(a_W)p_*^{|W|+2}}\\ &\qquad= \sum_{a,a_W\in A^{W+1}}\frac{\delta\log n}{p_*^{|W|+2}}\\ &\qquad\leq \frac{\delta|A|^{|W|+1}\log n}{p_*^{|W|+2}}\,,\end{aligned}$$

simultaneously for all $W\in\mathcal{B}_1$, eventually almost surely as $n\to\infty$. Then, if we take $c>\frac{32|A|}{p_*^{|V_n|+1}}$, we have that eventually almost surely as $n\to\infty$

$$\max_{W\in\mathcal{B}_1}\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}).$$

This completes the proof of part (a).
(b) We have to prove that

$$\max_{W\in\mathcal{B}_2}\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely as $n\to\infty$.

In this case we have that for all $W \in \mathcal{B}_2$

$$\begin{aligned}\mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}) - \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;=\; & \\ \sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} & N(a,a_{\mathrm{ne}(v)}) \log \frac{\hat{p}(a|a_{\mathrm{ne}(v)})}{\hat{p}(a|a_W)} - c\left(|A|^{|\mathrm{ne}(v)|} - |A|^{|W|}\right)\log n \\ = n \left[\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} \right. & \left. \frac{N(a,a_{\mathrm{ne}(v)})}{n} \log \frac{\hat{p}(a|a_{\mathrm{ne}(v)})}{\hat{p}(a|a_W)} - c\left(|A|^{|\mathrm{ne}(v)|} - |A|^{|W|}\right)\frac{\log n}{n} \right] .\end{aligned}$$

By the Strong Law of Large Numbers (Appendix A, Theorem 26) we have that

$$\begin{aligned}\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} & \frac{N(a,a_{\mathrm{ne}(v)})}{n} \log \frac{\hat{p}(a|a_{\mathrm{ne}(v)})}{\hat{p}(a|a_W)} \\ & \longrightarrow \sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} p(a,a_{\mathrm{ne}(v)}) \log \frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \end{aligned} \tag{5.7}$$

almost surely as $n \to \infty$. On the other hand,

$$c\left(|A|^{|\mathrm{ne}(v)|} - |A|^{|W|}\right)\frac{\log n}{n} \;\longrightarrow\; 0$$

when $n \to \infty$. Now, note that we can rewrite the right-hand side of (5.7) by

$$\begin{aligned}\sum_{(a,a_{\mathrm{ne}(v)})\in A^{\mathrm{ne}(v)+1}} & p(a,a_{\mathrm{ne}(v)}) \log \frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \\ & = \sum_{a_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}} p(a_{\mathrm{ne}(v)}) \sum_{a\in A} p(a|a_{\mathrm{ne}(v)}) \log \frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \\ & = \sum_{a_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}} p(a_{\mathrm{ne}(v)})\; D(p(.|a_{\mathrm{ne}(v)})\,;\, p(.|a_W))\,.\end{aligned}$$

By Lemma 23 (in Appendix A) and the minimality of $\mathrm{ne}(v)$ (see (5.1) and Lemmas 10 and 11) we must have $D(p(.|a_{\mathrm{ne}(v)})\,;\, p(.|a_W)) > 0$ for at least one $a_{\mathrm{ne}(v)}$, so

$$\sum_{a_{\mathrm{ne}(v)}\in A^{\mathrm{ne}(v)}} p(a_{\mathrm{ne}(v)})\; D(p(.|a_{\mathrm{ne}(v)})\,;\, p(.|a_W)) > 0$$

simultaneously for all $W \subset \mathrm{ne}(v)$, eventually almost surely as $n \to \infty$. Therefore

$$\max_{W\in\mathcal{B}_2} \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely as $n \to \infty$, completing the proof of case (b).
(c) In this case, we have to prove that

$$\max_{W\in\mathcal{B}_3} \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely as $n \to \infty$.

At first we will prove that

$$\max_{W\in\mathcal{B}_3} \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;\leq\; \mathrm{PML}(x_v^{(1:n)}|x_{V_n\setminus\{v\}}^{(1:n)})$$

eventually almost surely as $n \to \infty$. Note that we have

$$\begin{aligned}
&\mathrm{PML}(x_v^{(1:n)}|x_{V_n\setminus\{v\}}^{(1:n)}) - \mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \\
&= \sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} N(a,a_{V_n\setminus\{v\}}) \log \frac{\hat{p}(a|a_{V_n\setminus\{v\}})}{\hat{p}(a|a_W)} - c\left(|A|^{|V_n|-1} - |A|^{|W|}\right)\log n \\
&= n\left[\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \frac{N(a,a_{V_n\setminus\{v\}})}{n} \log \frac{\hat{p}(a|a_{V_n\setminus\{v\}})}{\hat{p}(a|a_W)} - c\left(|A|^{|V_n|-1} - |A|^{|W|}\right)\frac{\log n}{n}\right].
\end{aligned}$$

The second term in the brackets

$$c\left(|A|^{|V_n|-1} - |A|^{|W|}\right)\frac{\log n}{n} \longrightarrow 0$$

when $n \to \infty$. And, for the first term we have

$$\begin{aligned}
&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \frac{N(a,a_{V_n\setminus\{v\}})}{n} \log \frac{\hat{p}(a|a_{V_n\setminus\{v\}})}{\hat{p}(a|a_W)} \\
&= \sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \Big[\frac{N(a,a_{V_n\setminus\{v\}})}{n} \log \frac{\hat{p}(a|a_{V_n\setminus\{v\}})}{\hat{p}(a|a_W)} - \frac{N(a,a_{V_n\setminus\{v\}})}{n} \log p(a|a_W) \\
&+ \frac{N(a,a_{V_n\setminus\{v\}})}{n} \log p(a|a_W)\Big] \\
&= \sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \left[\frac{N(a,a_{V_n\setminus\{v\}})}{n} \log \frac{\hat{p}(a|a_{V_n\setminus\{v\}})}{p(a|a_W)} - \frac{N(a,a_{V_n\setminus\{v\}})}{n} \log \frac{\hat{p}(a|a_W)}{p(a|a_W)}\right]. \qquad (5.8)
\end{aligned}$$

We divide again the expression in two parts. Loking at the second term of the sum in (5.8) we have

$$\begin{aligned}
&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \frac{N(a,a_{V_n\setminus\{v\}})}{n} \log \frac{\hat{p}(a|a_W)}{p(a|a_W)} \\
&= \sum_{(a,a_W)\in A^{W+1}} \log \frac{\hat{p}(a|a_W)}{p(a|a_W)} \sum_{a_{V_n\setminus v\cup W}\in A^{V_n\setminus v\cup W}} \frac{N(a,a_{V_n\setminus\{v\}})}{n} \\
&= \sum_{(a,a_W)\in A^{W+1}} \frac{N(a,a_W)}{n} \log \frac{\hat{p}(a|a_W)}{p(a|a_W)} \\
&= \sum_{(a,a_W)\in A^{W+1}} \frac{N(a_W)}{n} \hat{p}(a|a_W) \log \frac{\hat{p}(a|a_W)}{p(a|a_W)} \\
&= \sum_{(a,a_W)\in A^{W+1}} \frac{N(a_W)}{n} D(\hat{p}(\cdot|a_W)\,;\, p(\cdot|a_W)).
\end{aligned}$$

By Lemma 24 (in Appendix A) and Corollary 21, we have that eventually almost surely

$$\begin{aligned}
&\max_{W\in\mathcal{B}_3} \sum_{(a,a_W)\in A^{W+1}} \frac{N(a_W)}{n} D(\hat{p}(\cdot|a_W)\,;\, p(\cdot|a_W)) \\
&\leq \max_{W\in\mathcal{B}_3} \frac{16|A|^{|W|+1}\log n}{np_*^{|W|+2}} \leq \frac{16|A|^{|V_n|}\log n}{np_*^{|V_n|+1}} \longrightarrow 0 \qquad (5.9)
\end{aligned}$$

as $n \to \infty$.

Now, as $\hat{p}(a|a_{V_n\setminus\{v\}})$ is the maximum likelihood estimator of $p(a|a_{V_n\setminus\{v\}})$ and V_n will eventually contain $\mathrm{ne}(v)$, the first term in the sum (5.8) can be lower-bounded by

$$\begin{aligned}
&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \frac{N(a,a_{V_n\setminus\{v\}})}{n}\log\frac{p(a|a_{V_n\setminus\{v\}})}{p(a|a_W)}\\
=&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \frac{N(a,a_{V_n\setminus\{v\}})}{n}\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)}\\
=&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \hat{p}(a,a_{V_n\setminus\{v\}})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)} \qquad (5.10)
\end{aligned}$$

By Theorem 20, $\hat{p}(a,a_{V_n\setminus\{v\}}) > p(a,a_{V_n\setminus\{v\}}) - \sqrt{\frac{\delta\log n}{n}}$. Ergo, for (5.10) we have

$$\begin{aligned}
&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \hat{p}(a,a_{V_n\setminus\{v\}})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)}\\
>&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \left[p(a,a_{V_n\setminus\{v\}}) - \sqrt{\frac{\delta\log n}{n}}\right]\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)}\\
=&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} p(a_{V_n\setminus\{v\}})p(a|a_{V_n\setminus\{v\}})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)}\\
-&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \sqrt{\frac{\delta\log n}{n}}\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)}\\
\geq&\sum_{(a_{V_n\setminus\{v\}})\in A^{V_n}-1} p(a_{V_n\setminus\{v\}})\sum_{a\in A} p(a|a_{\mathrm{ne}(v)})\log\frac{p(a|a_{\mathrm{ne}(v)})}{p(a|a_W)}\\
-&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \sqrt{\frac{\delta\log n}{n}}\log\frac{1}{p_{min}(v)}\\
=&\sum_{(a_{V_n\setminus\{v\}})\in A^{V_n}-1} p(a_{V_n\setminus\{v\}})D(p(\cdot|a_{\mathrm{ne}(v)})\,;\,p(\cdot|a_W))\\
+&\sum_{(a,a_{V_n\setminus\{v\}})\in A^{V_n}} \log p_{min}(v)\sqrt{\frac{\delta\log n}{n}} \qquad (5.11)
\end{aligned}$$

By Assumptions 14 and 17, (5.11) is grater than

$$\alpha + |A|^{|V_n|}\log p_{min(v)}\sqrt{\frac{\delta\log n}{n}} \geq \frac{\alpha}{2} > 0$$

eventually almost surely as $n\to\infty$, simultaneously for all $W\in\mathcal{B}_3$. So,

$$\max_{W\in\mathcal{B}_3}\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;\leq\; \mathrm{PML}(x_v^{(1:n)}|x_{V_n\setminus\{v\}}^{(1:n)})$$

eventually almost surely as $n\to\infty$.

As V_n will contain $\mathrm{ne}(v)$ eventually as $n\to\infty$, then $V_n\setminus\{v\}\in\mathcal{B}_1$, eventually as $n\to\infty$. Then, by item (a),

$$\mathrm{PML}(x_v^{(1:n)}|x_{V_n\setminus\{v\}}^{(1:n)}) \;\leq\; \max_{W\in\mathcal{B}_1}\mathrm{PML}(x_v^{(1:n)}|x_W^{(1:n)}) \;<\; \mathrm{PML}(x_v^{(1:n)}|x_{\mathrm{ne}(v)}^{(1:n)}).$$

So,

$$\max_{W \in \mathcal{B}_3} \mathrm{PML}(x_v^{(1:n)} | x_W^{(1:n)}) \; < \; \mathrm{PML}(x_v^{(1:n)} | x_{\mathrm{ne}(v)}^{(1:n)})$$

eventually almost surely as $n \to \infty$. This finishes the proof of case (c).

By combining the results of all three cases, we conclude that

$$\max_{W \subset V_n \setminus \{v\}, W \neq \mathrm{ne}(v)} \mathrm{PML}(x_v^{(1:n)} | x_W^{(1:n)}) \; < \; \mathrm{PML}(x_v^{(1:n)} | x_{\mathrm{ne}(v)}^{(1:n)})$$

and $\widehat{\mathrm{ne}}(v) = \mathrm{ne}(v)$, eventually almost surely as $n \to \infty$.

□

Proof of Corollary 19. The proof of the corollary follows from Theorem 18 and noting that V' is finite. The same arguments of the proof of Corollary 8 can be used here just changing the finite V by V', which is also finite. □

Chapter 6

Conclusion

In this thesis we introduced a new estimator for discrete Markov random fields on graphs, and we proved its consistency even for the case where the set of vertices is countable infinite (Chapter 5). In this case we highlight the theoretical difficulties and the differences between this and the case where the set of vertices is finite (Chapter 2). Some propositions had to be changed or even exchanged for other more powerful results. And even for the case where the set of vertices is countable infinite, we did not need to assume that the model satisfies the usual "positivity" condition.

To estimate the graph estructure, i.e. to estimate the set of edges E, we first estimate for each vertex its neighborhood, based on a penalized maximum condicional likelihood criterion. We showed that this estimator converges almost surely to the true set of neighbors. Then we showed that, combing the estimators of all vertices, the estimator of $G = (V, E)$ converges almost surely to the true graph. When we are dealing with a countable infinite set of vertices we only estimate a finite subgraph of G, in this case the estimator converges almost surely to the true finite subgraph.

The results obtained by the simulations in Chapter 3 corroborated the theoretical statements about the consistency of the estimator. We were able to evaluate the performance of the estimator from different angles: considering the sample size, the value of the constant c of the estimator, and the choice between a conservative or non conservative approach. Through these evaluations we were able to conclude on the superiority of the non conservative approach. Also we were able to see the impact caused by the value of c, transforming the estimator in more or less conservative. The higher the value of c, the more conservative is the estimator. On the other hand, for fixed c, we observed the convergence of the estimated graphs when the sample size grows, confirming the consistency of the estimator. Other factors related to the complexity of the model influence the estimation, as the number of vertices, the alphabet size and the dependence structure of the model (represented by the graph). Few vertices, small alphabets, and graphs with few edges are easier to estimate, in the sense that require smaller samples and are computationally less expensive.

We can apply this model to several real data, especially when we are interested in discovering the conditional dependence structure of the variables involved. Of course, some considerations must be made, our model deals with discrete variables on finite alphabets, and to estimate the models the samples must have independent observations. When in real data the variables are continuous we can discretized them, and this discretization should make sense for the application. But a more difficult problem to handle with is the lack of independence between the sample observations. To overcome this issue some methods can be used, as for example reduce the sample taking only observations separated by a gap. This gap can be choosen based on the autocorrelation function, as in the case of EEG signals presented in this thesis.

In many cases of real data applications we have a lot of variables (represented in the graph by its vertices) and sample sizes that are not large enough. In these cases all information about the absence of edges should be used to simplify the problem. Because we estimate the set of neighbors of each vertex, this information can be used in a more simple way. And the program used here to estimate the model has the option of inciating the algoritm with an adjacency matrix different from the complete graph.

In the applications we introduced an empirical method to discover the best value of the constant c, based on k-fold cross validation, although other methods are also possible, as for example the bootstrap. Though, the question of how to choose the best constant c remains open, and we intend to continue to work on it in the future.

Appendix A

Basic probability results

Definition 22. (Küllback-Leibler divergence) *The Küllback-Leibler divergence between the two probabilities distribution P and Q over A is defined by*

$$D(P;Q) = \sum_{a \in A} P(a) \log \frac{P(a)}{Q(a)}$$

where, by convention, $P(a) \log \frac{P(a)}{Q(a)} = 0$ if $P(a) = 0$ and $P(a) \log \frac{P(a)}{Q(a)} = +\infty$ if $P(a) > Q(a) = 0$.

Lemma 23. *For the Küllback-Leibler divergence, we have that $D(P;Q) \geq 0$ and $D(P;Q) = 0$ if and only if $P(a) = Q(a)$ $\forall a \in A$.*

The following lemma was taken from Csiszár and Talata (2006, Lemma 6.3).

Lemma 24. *For any P and Q we have*

$$D(P;Q) \leq \sum_{a \in A:\ Q(a)>0} \frac{[P(a) - Q(a)]^2}{Q(a)}\,.$$

Definition 25. (Empirical mean) *Let $X_1, X_2, \ldots, X_n$ be random variables, the empirical mean ($\bar{X}$) of this variables is defined by*

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i.$$

This next theorem can be found in Breiman (1968, Section 3.9).

Theorem 26. (Law of the Iterated Logarithm) *Let $Y_1, Y_2, \ldots$ be independent identically distributed random variables with $\mathbb{E}[Y_i] = 0$ and $\mathbb{E}[Y_i^2] < \infty$, and $Z_n = \sum_{i=1}^{n} Y_i$ then*

$$\limsup_{n \to \infty} \frac{|Z_n|}{\mathbb{E}[Y_i^2]\sqrt{2n \log\log n}} = 1 \text{ almost surelly.}$$

And this theorem can be found in Breiman (1968, Section 3.6).

Theorem 27. (Strong Law fo Large Numbers) *Let $X_1, X_2, \ldots$ be independent identically distributed random variables with $\mathbb{E}[X_i] = \mu$ and $\mathbb{E}[|X_i|] < \infty$, then*

$$\bar{X} \to \mu \text{ almost surelly as } n \to \infty.$$

The next theorem and its proof can be found in Hoeffding (1963).

Theorem 28. (Hoeffding's inequality) *Let $X_1, \ldots, X_n$ be independent random variables where X_i are strictly bounded by the intervals $[a_i, b_i]$. Then, for all $t > 0$*

$$\mathbb{P}(|\bar{X} - \mathbb{E}[\bar{X}]| \geq t) \leq \exp\left(-\frac{2n^2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$

Appendix B

R Programs

B.1 Generating samples

Here are presented the R programs that genarete samples from the examples in Chapter 2, whose algorithms are described in Section 3.1 of Chapter 3. The following programs can generate samples of size n for each Markov random field given as examples. Remember that because the theoretical probabilities described in Chapter 2 are fixed, they are not parameters of the created function.

B.1.1 Example 1

The following program generates a sample of size n of the Markov random field in Example 1 (Section 3.1), with 5 variables in the aphabet $A = (0, 1, 2)$.

```
sample_graph_1<-function(n){
    A<-c(0,1,2)
    x<-matrix(rep(0,times=n*5),ncol=5,nrow=n)

    # matrix for the conditional probability of X1 given X3
    # (X3 is the rows and X1 is the columns)
    p1_3<-matrix(c(2,4,4,3,4,3,4,3,3)/10, nrow=3, ncol=3, byrow=T)

    # matrix for the conditional probablitily of X2 given X1 and X3
    p2_13<-matrix( c(1/2,1/2,0,2/4,1/4,1/4,1/4,1/4,2/4,1/3,0,2/3,1/4,
    1/4,2/4,1/3,2/3,0,0,3/4,1/4,1/3,1/3,1/3,1/3,1/3,1/3),
    nrow=9, ncol=3, byrow=T)

    # matrix for the conditional probability of X4 given X3
    # (X3 is the rows and X4 is the columns)
    p4_3<-matrix(c(1,4,5,2,7,1,3,6,1)/10, nrow=3, ncol=3, byrow=T)

    # matrix for the conditional probability of X5 given X3
    # (X3 is the rows and X5 is the columns)
    p5_3<-matrix(c(2,6,2,3,1,6,4,3,3)/10, nrow=3, ncol=3, byrow=T)

    # probability of X3
    p3=c(0.3,0.2,0.5)

    x[,3]<-sample(A,size=n,prob=p3,replace=TRUE)

    for(i in 1:n){
```

```
        x[i,1]<-sample(A,size=1,prob=p1_3[x[i,3]+1,])
        x[i,4]<-sample(A,size=1,prob=p4_3[x[i,3]+1,])
        x[i,5]<-sample(A,size=1,prob=p5_3[x[i,3]+1,])
        x[i,2]<-sample(A,size=1,prob=p2_13[3*x[i,3]+x[i,1]+1,])
    }

    return(x)
}
```

B.1.2 Example 2

The following program generates a sample of size n of the Markov random field in Example 2 (Section 3.1), with 7 variables in the aphabet $A = (0, 1)$.

```
sample_graph_2<-function(n){
    A<-c(0,1)
    x<-matrix(rep(0,times=7*n),ncol=7,nrow=n)

    # probabilities of variables X3 and X6
    p3=c(.4,.6)
    p6=c(.6,.4)

    # matrix for the conditional probability of X1 given X2
    # (X2 is the rows and X1 is the columns)
    p1_2<-matrix(c(.3,.7,.6,.4),ncol=2,nrow=2)

    # matrix for the conditional probability of X2 given X3
    # and X4 given X3
    # (X3 is the rows and X2 and X4 are the columns)
    p2_3<-matrix(c(.4,.6,.5,.5),ncol=2,nrow=2)
    p4_3<-matrix(c(.7,.3,.2,.8),ncol=2,nrow=2)

    # matrix for the conditional probability of X5 given X6
    # and X7 given X6
    # (X6 is the rows and X5 and X7 are the columns)
    p5_6<-matrix(c(0.3,.7,.8,.2),ncol=2,nrow=2)
    p7_6<-matrix(c(0.5,.5,.3,.7),ncol=2,nrow=2)


    x[,3]<-sample(A,size=n,prob=p3,replace=T)
    x[,6]<-sample(A,size=n,prob=p6,replace=T)

    for(i in 1:n){
        x[i,2]<-sample(A,size=1,prob=p2_3[x[i,3]+1,])
        x[i,4]<-sample(A,size=1,prob=p4_3[x[i,3]+1,])
        x[i,5]<-sample(A,size=1,prob=p5_6[x[i,6]+1,])
        x[i,7]<-sample(A,size=1,prob=p7_6[x[i,6]+1,])
        x[i,1]<-sample(A,size=1,prob=p1_2[x[i,2]+1,])
    }

    return(x)
}
```

B.1.3 Example 3

The following program generates a sample of size n of the Markov random field in Example 3 (Section 3.1), with 12 variables in the aphabet $A = (0, 1, 2)$.

```
sample_graph_3<-function(n){
    A<-c(0,1,2)
    x<-matrix(rep(0,times=12*n),ncol=12,nrow=n)

    # initial probability and transition matrix of the Markov Chain
    p1<-c(.3,.6,.1)
    P<-matrix(c(.2,.5,.3,.7,.2,.1,.4,.3,.3),ncol=3,nrow=3)

    x[,1]<-sample(A,size=n,prob=p1,replace=T)

    for(i in 1:n){
        for(j in 2:12){
            x[i,j]<-sample(A,size=1,prob=P[x[i,j-1]+1,])
        }
    }

    return(x)
}
```

B.1.4 Example 4

The following program generates a sample of size n of the Markov random field in Example 4 (Section 3.1), with 10 variables in the aphabet $A = (0, 1, 2)$.

```
sample_graph_4<-function(n){
    A<-c(0,1,2)
    x<-matrix(rep(0,times=10*n),ncol=10,nrow=n)

    # probabilities of variables X4 and X10
    p4<-c(.3,.4,.3)
    p10<-c(.4,.1,.5)

    # matrix for the conditional probability of X1 given X2
    # (X2 is the rows and X1 is the columns)
    p1_2<-matrix(c(.2,.3,.5,.4,.3,.3,.3,.5,.1),ncol=3,nrow=3)

    # matrix for the conditional probability of X2 given X4,
    # X3 given X4, X5 given X4 and X6 given X4
    # (X4 is the rows and X2, X3, X5 e X6 are the columns)
    p2_4<-matrix(c(.4,.4,.2,.5,.2,.3,.5,.3,.2),ncol=3,nrow=3)
    p3_4<-matrix(c(.2,.6,.2,.3,.3,.4,.4,.2,.4),ncol=3,nrow=3)
    p5_4<-matrix(c(.7,.2,.1,.3,.5,.2,.1,.5,.4),ncol=3,nrow=3)
    p6_4<-matrix(c(.3,.2,.5,.6,.2,.2,.4,.3,.3),ncol=3,nrow=3)

    # matrix for the conditional probability of X7 given X10,
    # and X9 given X10
    # (X10 is the rows and X7 e X69 are the columns)
    p7_10<-matrix(c(.3,.3,.4,.4,.2,.4,.5,.3,.2),ncol=3,nrow=3)
    p9_10<-matrix(c(.4,.4,.2,.3,.4,.3,.5,.2,.3),ncol=3,nrow=3)
```

```
    # matrix for the conditional probability of X8 given X9 and X10,
    # (X9 and X10 are the rows and X8 is the columns)
    p8_910<-matrix(c(3/4,0,1/4,1/4,2/4,1/4,1/2,1/2,0,0,
    2/3,1/3,2/4,1/4,1/4,1/3,1/3,1/3,1/5,2/5,2/5,1/2,0,
    1/2,1/3,1/3,1/3),ncol=3,nrow=9)

    x[,4]<-sample(A,size=n,pro=p4,replace=TRUE)
    x[,10]<-sample(A,size=n,pro=p10,replace=TRUE)

    for(i in 1:n){
        x[i,2]<-sample(A,size=1,prob=p2_4[x[i,4]+1,])
        x[i,3]<-sample(A,size=1,prob=p3_4[x[i,4]+1,])
        x[i,5]<-sample(A,size=1,prob=p5_4[x[i,4]+1,])
        x[i,6]<-sample(A,size=1,prob=p6_4[x[i,4]+1,])
        x[i,1]<-sample(A,size=1,prob=p1_2[x[i,2]+1,])
        x[i,7]<-sample(A,size=1,prob=p7_10[x[i,10]+1,])
        x[i,9]<-sample(A,size=1,prob=p9_10[x[i,10]+1,])
        x[i,8]<-sample(A,size=1,prob=p8_910[3*x[i,10]+x[i,9]+1,])
    }

    return(x)
}
```

B.2 Programs that measure the estimator performance

Here are presented the R programs that were used to measure the estimator performance in Chapter 3. Because the adjacency matrix is simetric, it is possible to represent those matrices using vectors in such a way that a $p \times p$ matrix is represented by a vector of length $0.5p(p-1)$. This next example tries to explain how this tranformation works, remember that in the adjacency matrix all the numbers on the diagonal are equal to zero and that it is symetric, then:

$$\begin{bmatrix} 0 & a & b & c & d \\ a & 0 & e & f & g \\ b & e & 0 & h & i \\ c & f & h & 0 & j \\ d & g & i & j & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} a & b & c & d & e & f & g & h & i & j \end{bmatrix}.$$

So this next R program is a function that transforms an adjacency matrix into a vector:

```
matrix_to_vector<-function(m){
    p<-dim(m)[1]
    v<-m[1,2:p]
    for(i in 2:(p-2)){
        v<-c(v,m[i,(i+1):p])
        i<-i+1
    }
    v<-c(v,m[p-1,p])
    return(v)
}
```

The next R program calculates the underestimation, overestimation and total errors given the true graph and an estimative for this graph. Both have to be represented by vectors, not by their adjacency matrices. The parameter `g` is the vector that represents the real graph, and `eg` is the vector

that represents the estimated graph. The function returns the underestimation, the overestimation and the total errors, in this order.

```
errors<-function(g,eg){
    n<-length(g)
    error.under<-0
    error.over<-0

    for(i in 1:n){
        if(g[i]!=eg[i]){
            if(g[i]<eg[i]){
                error.over<-error.over+1
            }
            else{
                error.under<-error.under+1
            }
        }
        i<-i+1
    }
    error.total<-(error.under+error.over)/n
    error.under<-error.under/sum(g)
    error.over<-error.over/(n-sum(g))

    results<-cbind(error.under,error.over,error.total)
    return(results)
}
```

Now, this last R program is a function that gives the pair (false positive rate, true positive rate) = (1− specificity, sensibility) of the ROC curve. Again the vectores are used instead of the adjacency matrices. And the parameter `g` represents the real graph vector and `eg` represents the estimated graph vector.

```
roc.curve<-function(g,eg){
    n<-length(g)

    confusion_matrix<-cbind(c(0,0),c(0,0))
    for(j in 1:n){
        if(eg[j]==g[j]){
            if(g[j]==1){
                confusion_matrix[1,1]<-confusion_matrix[1,1]+1
            }
            else{
                confusion_matrix[2,2]<-confusion_matrix[2,2]+1
            }
        }
        else{
            if(g[j]==1){
                confusion_matrix[1,2]<-confusion_matrix[1,2]+1
            }
            else{
                confusion_matrix[2,1]<-confusion_matrix[2,1]+1
            }
        }
```

```
    }
    tp.rt<-(confusion_matrix[1,1]/sum(confusion_matrix[1,])
    fp.rt<-(confusion_matrix[2,1]/sum(confusion_matrix[2,])

    result<-cbind(fp.rt,tp.rt)
    return(result)
}
```

B.3 Log likelihood for cross-validation

The following R program is a function that calculates the log likelihood needed in the cross-validation method (Chapter 4). Its parameters are: `m.adj` that represents the adjacency matrix of neighbors estimated using the training sample, `s.training` the training sample, and `s.test` the test sample.

```
log_likelihood<-function(m.adj,s.training,s.test){
    I<-dim(m.adj)[1]
    log.like<-0
    num.nei<-rowSums(m.adj)
    for(i in 1:I){
        aux<-0
        nei.v.training<-0
        nei.v.test<-0
        if(num.nei[i]>0){
            for(j in 1:I){
                if(m.adj[i,j]==1){
                    if(aux==0){
                        nei.v.training<-s.training[,j]
                        nei.v.test<-s.test[,j]
                    }
                    else{
                        nei.v.training<-paste0(nei.v.training,
                                               s.training[,j])
                        nei.v.test<-paste0(nei.v.test,s.test[,j])
}
                    aux<-aux+1
                }
            }
            t.training<-table(s.training[,i],nei.v.training)
            t.test<-table(s.test[,i],nei.v.test)
            probs<-matrix(rep(0,times=dim(t.training)[1]
                         *dim(t.training)[2]),ncol=dim(t.training)[2])
            for(r.p in 1:dim(probs)[1]){
                for(c.p in 1:dim(probs)[2]){
                    probs[r.p,c.p]<-t.training[r.p,c.p]/
                                    sum(t.training[,c.p])
                }
            }

            if(dim(t.training)[2]!=dim(t.test)[2]){
                n.training<-colnames(t.training)
```

```
            n.test<-colnames(t.test)
            dif<-dim(t.training)[2]-dim(t.test)[2]
            k<-l<-1
            d<-0
            while(d<dif){
                if(n.training[k]==n.test[l]){
                    k<-k+1
                    l<-l+1
                }
                else{
                    d<-d+1
                    n.training<-n.training[-k]
                    probs<-probs[,-k]
                }
            }

        }
        for(r in 1:dim(t.test)[1]){
            for(c in 1:dim(t.test)[2]){
                if(t.test[r,c]!=0 & probs[r,c]!=0){
                    log.like<-log.like+t.test[r,c]*log(probs[r,c])
                c<-c+1
            }
            r<-r+1
        }
    }
    else{
        t.training<-table(s.training[,i])
        t.test<-table(s.test[,i])
        probs<-t.training/sum(t.training)

        for(r in 1:length(t.test)){
            if(t.test[r]!=0 & probs[r]!=0){
                log.like<-log.like+t.test[r]*log(probs[r])
            }
            r<-r+1
        }
    }
  }
  return(log.like)
}
```

Bibliography

Leo Breiman. *Probability*. Addison-Wesley, Reading, Massachusets, first edition, 1968. 53

P. Collinson. Of bombers, radiologists, and cardiologists: time to ROC. *Heart*, 80(3):215–217, 1998. 31

Imre Csiszár. Large-scale typicality of Markov sample paths and consistency of MDL order estimators. *IEEE Trans. Inform. Theory*, 48(6):1616–1628, 2002. ISSN 0018-9448. doi: 10.1109/TIT.2002.1003842. URL http://dx.doi.org/10.1109/TIT.2002.1003842. Special issue on Shannon theory: perspective, trends, and applications. 43

Imre Csiszár and Zsolt Talata. Consistent estimation of the basic neighborhood of Markov random fields. *Ann. Statist.*, 34(1):123–145, 2006. ISSN 0090-5364. doi: 10.1214/009053605000000912. URL http://dx.doi.org/10.1214/009053605000000912. 1, 4, 41, 53

Tom Fawcett. An Introduction to ROC Analysis. *Pattern Recogn. Lett.*, 27(8):861–874, jun 2006. ISSN 0167-8655. doi: 10.1016/j.patrec.2005.10.010. URL http://dx.doi.org/10.1016/j.patrec.2005.10.010. 22

Antonio Galves, Enza Orlandi, and Daniel Y. Takahashi. Identifying interacting pairs of sites in Ising models on a countable set. *Braz. J. Probab. Stat.*, 29(2):443–459, 2015. 1

Hans-Otto Georgii. *Gibbs measures and phase transitions*, volume 9 of *de Gruyter Studies in Mathematics*. Walter de Gruyter & Co., Berlin, second edition, 2011.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition, 2009. URL http://www-stat.stanford.edu/~tibs/ElemStatLearn/. 13, 35

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):pp. 13–30, 1963. ISSN 01621459. URL http://www.jstor.org/stable/2282952. 53

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: With Applications in R*. Springer Publishing Company, Incorporated, 2014. ISBN 1461471370, 9781461471370. 35

Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009. ISBN 0262013193, 9780262013192. 1

Steffen L. Lauritzen. *Graphical Models*. Oxford University Press, 1996. 1, 4

Po-Ling Loh and Martin J. Wainwright. Structure estimation for discrete graphical models: Generalized covariance matrices and their inverses. *Ann. Statist.*, 41(6):3022–3049, 2013. 1

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *Ann. Statist.*, 34(3):1436–1462, 2006. 1

Pradeep Ravikumar, Martin J. Wainwright, and John D. Lafferty. High-dimensional Ising model selection using l_1-regularized logistic regression. *Ann. Statist.*, 38(3):3022–1319, 2010. 1

Benjamin Reiser and David Faraggi. Confidence Intervals for the Generalized ROC Criterion. *Biometrics*, 53(2):644–652, 1997. ISSN 0006341X, 15410420. URL http://www.jstor.org/stable/2533964. 31

Maria L. Rizzo. *Statistical Computing whit R*, volume 9 of *Computer Science and Data Analysis Series*. Chapman & Hall/CRC, first edition, 2007. 20

G. Schwarz. Estimating the dimension of a model. *Ann. Statist.*, 6:461–464, 1978. 1, 3, 13

A. Shojaie and G. Michailidis. Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs. *Biometrika*, 97(3):519–538, 2010. 1

David Strauss and Michael Ikeda. Pseudolikelihood Estimation for Social Networks. *Journal of the American Statistical Association,*, 85(409):204–212, 1990. 1

R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1): 267–288, 1996. 1

C. S. Wallace and D. M. Boulton. An information measure for classification. *Computer Journal*, 11(2):185–194, 1968. 13

M. H. Zweig and G Campbell. Receiver-operating characteristic (ROC) plots: a fundamental evaluation toll in clinical medicine. *Clinical chemistry*, 39(4):561–577, 1993. 30