EXPECTATION MAXIMIZATION AND LATENT CLASS MODELS

By

Hector Banos

RECOMMENDED:

Dr. Edward Bueler

Dr. Ron Barry

Dr. Jill Faudree

Dr. Elizabeth Allman
Advisory Committee Chair

Dr. John Rhodes
Chair, Department of Mathematics and Statistics

APPROVED:

Dr. Paul Layer
Dean, College of Natural Science and Mathematics

Dr. John Eichelberger
Dean of the Graduate School

25 April 2016
Date

EXPECTATION MAXIMIZATION AND LATENT CLASS MODELS

A

THESIS

Presented to the Faculty

of the University of Alaska Fairbanks

in Partial Fulfillment of the Requirements

for the Degree of

MASTER OF SCIENCE

By

Hector Banos, B.S.

Fairbanks, Alaska

May 2016

Abstract

Latent tree models are tree structured graphical models where some random variables are observable while others are latent. These models are used to model data in many areas, such as bioinformatics, phylogenetics, computer vision among others. This work contains some background on latent tree models and algebraic geometry with the goal of estimating the volume of the latent tree model known as the 3-leaf model $\mathcal{M}_2$ (where the root is a hidden variable with 2 states, and is the parent of three observable variables with 2 states) in the probability simplex Δ_7, and to estimate the volume of the latent tree model known as the 3-leaf model $\mathcal{M}_3$ (where the root is a hidden variable with 3 states, and is the parent of two observable variables with 3 states and one observable variable with 2 states) in the probability simplex Δ_{17}. For the model $\mathcal{M}_3$, we estimate that the rough percentage of distributions that arise from stochastic parameters is 0.015%, the rough percentage of distributions that arise from real parameters is 64.742% and the rough percentage of distributions that arise from complex parameters is 35.206%. We will also discuss the algebraic boundary of these models and we observe the behavior of the estimates of the Expectation Maximization algorithm (EM algorithm), an iterative method typically used to try to find a maximum likelihood estimator.

Table of Contents

Page

Signature Page . . . . . **i**

Title Page . . . . . **iii**

Abstract . . . . . **v**

Table of Contents . . . . . **vii**

List of Figures . . . . . **ix**

List of Tables . . . . . **xi**

Acknowledgments . . . . . **xiii**

Chapter 1: Introduction **1**

1.1 Chapter Overview 1

Chapter 2: Basic Concepts **3**

2.1 A statistical model as a geometric object 3

2.2 Varieties 4

2.2.1 Zariski closure 5

2.2.2 Semialgebraic sets 6

2.3 Tensors 6

2.4 Latent tree models 8

2.4.1 Basic graph theory 8

2.4.2 Latent tree model 8

Chapter 3: The 3-leaf model **11**

3.0.3 Parameter identifiability 16

3.1 Volume of the model in the probability simplex 18

3.1.1 The volume of $\mathcal{M}_2$ in Δ_7 19

3.1.2 The volume of $\mathcal{M}_3$ in Δ_{17} 19

Chapter 4: The algebraic boundary of $\mathcal{M}$ 21

4.1 Algebraic boundary 21

4.2 Boundary strata of $\mathcal{M}_2$ 23

4.3 Boundary strata of $\mathcal{M}_3$ 24

Chapter 5: The EM algorithm 27

5.1 Maximum likelihood estimator 27

5.2 The EM algorithm 29

5.3 E-step 29

5.4 M-step 30

5.5 EM estimates 30

Chapter 6: Conclusions 37

Appendix 38

References 55

List of Figures

Page

Figure 1.1 S_3: The directed tree associated to the model $\mathcal{M}$ 2

Figure 2.1 The probability simplex Δ_2 in $\mathbb{R}^3$ 4
Figure 2.2 Directed tree with root R, inner vertex A, and leaves X, Y and Z. . . 9

Figure 3.1 S_3: The directed tree associated to the model $\mathcal{M}$ 11

List of Tables

Page

Table 4.1 Boundary strata of $\mathcal{M}_2$ distributions generically of rank 2 24

Table 4.2 Boundary strata of $\mathcal{M}_3$ distributions generically of rank 3 26

Table 5.1 EM estimate attracted to various boundary strata of $\mathcal{M}_2$ for 10^4 random probability distributions. 33

Table 5.2 EM estimate attracted to various boundary strata of $\mathcal{M}_3$ for 10^5 random probability distributions. 34

Acknowledgments

I would like to express the deepest appreciation to my advisor Dr. Elizabeth Allman for letting me work with her, for introducing me to this fantastic topic, for her patience and for contributing to this work.

I would also like to express appreciation to my committee, Dr. John Rhodes, Dr. Edward Bueler, Dr. Jill Faudree and Dr. Ron Barry for taking time to review and give me suggestions for this work.

In addition, I would like to thank to Dr. David Maxwell, Dr. Margaret Short and Dr. Castaño for taking some time to answer questions and help me. Also, I would like to thank Tony Knowles for carefully reading and suggesting revisions.

Chapter 1

Introduction

A parametric model is a collection of probability distributions such that any element can be described by a finite dimensional vector of parameters θ in a parameter space Θ via a map (called a parametrization or parametric map) $\psi : \Theta \to \Delta_{m-1}$, where Δ_{m-1} denotes the probability simplex in $\mathbb{R}^m$. Particular types of parametric models are the latent tree models; these are rooted graphical models where some random variables are observable and some are latent (hidden). The 3-leaf model is a latent tree model with one latent variable as a root that is the parent of three conditionally independent observable variables. This model can be described semialgebraically, and in this thesis we work to understand better two particular 3-leaf models. For instance, we estimate the volume of the set of probability distributions for these models in their corresponding probability simplex and see the behavior of the estimates from the EM algorithm.

1.1 Chapter Overview

In chapter 2 we define some basic concepts of algebraic geometry and statistics. In particular, we talk about parametric models, varieties, semialgebraic sets, tensors and latent tree models.

In chapter 3, we introduce the 3-leaf latent tree model $\mathcal{M}$ with tree shown in figure 1.1. We describe some of its particular properties. We also discuss two particular models, $\mathcal{M}_2$ and $\mathcal{M}_3$, that are particularly interesting due to a particular property they share; for $i = 2, 3$ the Zariski closure of $\mathcal{M}_i$ intersected with the corresponding probability simplex is the whole probability simplex $\overline{\mathcal{M}_i} \cap \Delta_{q-1} = \Delta_{q-1}$. Our discussion includes parameter identifiability. We then find the estimate of the volume of $\mathcal{M}_2$ in the probability simplex Δ_7 [ZS12] and we compute our estimate of the volume of model $\mathcal{M}_3$ in the probability simplex Δ_{17}.

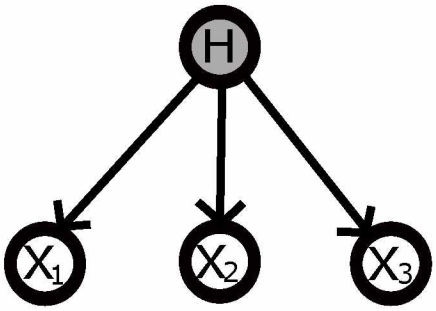

Figure 1.1: S_3: The directed tree associated to the model $\mathcal{M}$

In chapter 4, we describe the geometry of our models $\mathcal{M}_2$ and $\mathcal{M}_3$ and in particular we review and find the algebraic boundary of each of these models. We characterize the irreducible components of the algebraic boundary and identify each with some properties of the parameters. This will help to understand the estimates of the EM algorithm in chapter 5.

In chapter 5, we review the maximum likelihood function and the EM algorithm. We explain the EM algorithm and some properties of it in our particular case. We also show some of the results we obtain with the estimates of EM using the theory discussed in chapter 4.

Chapter 2

Basic Concepts

2.1 A statistical model as a geometric object

Let X be a discrete random variable with values in a finite set $\mathcal{X}$. If $\mathcal{X}$ has m elements, then without loss of generality we assume that $\mathcal{X} = \{1, \ldots, m\}$ and we identify the probability distribution of X (also known as a multinomial distribution) with a point $p = (p_1, \ldots, p_m) \in \mathbb{R}^m$ such that $p_x \geq 0$ for every $x \in \mathcal{X}$ and $\sum_{x \in \mathcal{X}} p_x = 1$. We define the *probability simplex* as the set of all such points

$$\Delta_{m-1} := \{p \in \mathbb{R}^m : p_x \geq 0, \sum_{x \in \mathcal{X}} p_x = 1\}. \tag{2.1}$$

A statistical model is a family of probability distributions and hence a family of points in Δ_{m-1}. This allows us to identify discrete statistical models with a geometric object.

A *parametric model* is a collection of probability distributions such that any element of this collection can be described by a finite dimensional vector of parameters $\theta \in \Theta$, where Θ is known as the *parameter space* (the space of all the possible values of parameters). In this case there is a map $\psi : \Theta \mapsto \Delta_{m-1}$ such that the model is equal to the image of Θ under ψ. The coordinates of this map are typically denoted by $\psi_x(\theta)$ for $x \in \mathcal{X}$ and $\theta \in \Theta$.

Example 1. Figure 2.1 shows the probability simplex Δ_2 in $\mathbb{R}^3$ where Δ_2 is the object defined by $1 = x + y + z$, $0 \leq x$, $0 \leq y$ and $0 \leq z$.

We are particularly interested in a parametric model known as a *latent tree model*, but before we define this type of model we introduce some basic concepts and definitions.

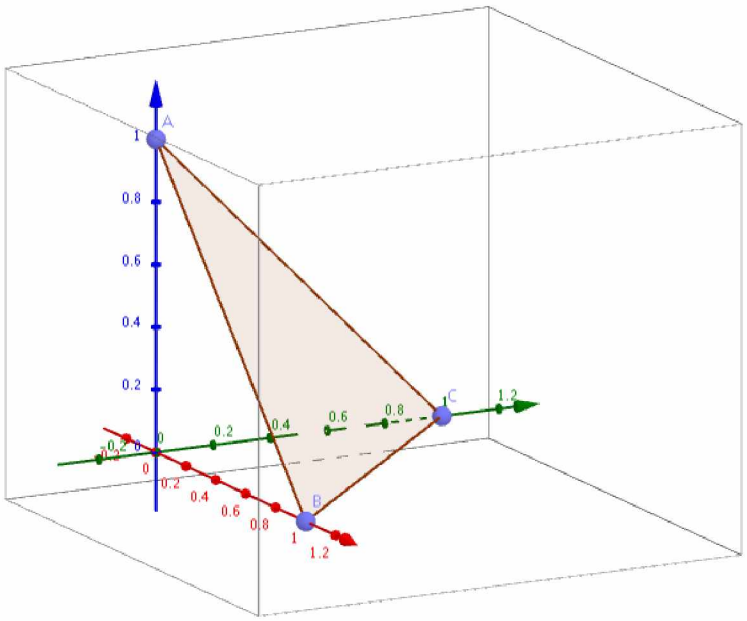

Figure 2.1: The probability simplex Δ_2 in $\mathbb{R}^3$

2.2 Varieties

Consider n indeterminates $\mathbf{x} = (x_1, .., x_n)$. Recall that a *polynomial* in $\mathbf{x}$ is any sum of the form

$$f(\mathbf{x}) = \sum_{\alpha_1=0}^{\infty} \cdots \sum_{\alpha_n=0}^{\infty} c_{\alpha_1 \cdots \alpha_n} x_1^{\alpha_1} \cdots x_n^{\alpha_n}, \ c_{\alpha_1 \cdots \alpha_n} \in \mathbb{R}$$

such that only finite number of $c_\alpha = c_{\alpha_1 \cdots \alpha_n}$ are non-zero. We can express it in its compact form $f(\mathbf{x}) = \sum_\alpha c_\alpha \mathbf{x}^\alpha$. We call c_α the *coefficient* of $\mathbf{x}^\alpha$.

Every polynomial $f = \sum_\alpha c_\alpha \mathbf{x}^\alpha$ defines a function $f : \mathbb{R}^n \to \mathbb{R}$, which is called a *polynomial function*. We make the distinction between a polynomial as an algebraic object and the map it defines. This distinction will allows us to link algebra and geometry by relating a polynomial f and the collection of zeros of the map it defines.

We denote the set of all polynomials in $x_1, .., x_n$ with coefficients in a field K by $K[\mathbf{x}]$ or $K[x_1, ..., x_n]$. This set forms a commutative ring with standard addition and multiplication of polynomials, with 0 denoting the zero polynomial. In this work we will be working only with the fields $\mathbb{C}$ and $\mathbb{R}$.

Definition 2. Let $f_1, ..., f_s$ be polynomials in $K[x_1, ..., x_n]$. The *affine algebraic variety* defined by $f_1, ..., f_s$ is

$$\mathcal{V}_K(f_1, ..., f_s) := \{(a_1, .., a_n) \in K^n : f_i(a_1, ..., a_n) = 0 \text{ for all } 1 \leq i \leq s\}.$$

Thus an affine variety $\mathcal{V}_K(f_1, ..., f_s)$ is the set of all solutions in K^n to the system of equations

$$f_1(x_1, ..., x_n) = \cdots = f_s(x_1, ..., x_n) = 0$$

Example 3. Consider the polynomials $f(x_1, x_2, x_3) = x_1 + x_2 + x_3 - 1$, $g(x_1, x_2, x_3) = x_1x_2$, $h(x_1, x_2, x_3) = x_2x_3$ and $k(x_1, x_2, x_3) = x_1x_3$. Then $\mathcal{V}_{\mathbb{R}}(f, g, h, k) = \{(1, 0, 0), (0, 0, 1), (0, 1, 0)\}$.

The image of the parametrization map of a latent tree model we consider defines a dense subset of a variety, and as previously mentioned this will allow us to link the algebra and the geometry. The next subsection defines a concept toward this goal.

2.2.1 Zariski closure

Let $S \subset K^n$. We define the *Zariski closure* $\overline{S}$ *of* S as the smallest algebraic variety in K^n containing S.

Definition 4. Let $S \subset K^n$ be any subset. We define the *ideal of* S by

$$\mathcal{I}(S) := \{f \in K[x_1, .., x_n] : f(a_1, .., a_n) = 0 \text{ for all } (a_1, .., a_n) \in S\}.$$

It is easy to see that this is an ideal in the ring of polynomials.

Proposition 5. *If $S \subset K^n$, the affine algebraic variety $\mathcal{V}_K(\mathcal{I}(S))$ is the Zariski closure $\overline{S}$.*

The proof of this proposition can be found in [CLO07].

Example 6. Consider the set $(0, 1) \subset \mathbb{R}$. Observe that the closure of $(0, 1)$ on the usual topology of $\mathbb{R}$ is the set $[0, 1]$. Now we observe that the Zariski closure of $(0, 1)$ is $\mathbb{R}$. This can be proved by showing that if a polynomial is zero in any nonempty open set of $\mathbb{R}$ (in the usual topology), then the polynomial is the zero polynomial.

2.2.2 Semialgebraic sets

Definition 7. A *basic semialgebraic set* is a subset S of $\mathbb{R}^n$ defined by polynomial equations and inequalities. More formally, S is a set of the form

$$S = \left(\bigcap_{i=1}^{r}\{\mathbf{x} \in \mathbb{R}^n : f_i(x) < 0\}\right) \bigcap \left(\bigcap_{j=r+1}^{s} \{\mathbf{x} \in \mathbb{R}^n : f_j(x) = 0\}\right)$$

where $f_i, f_j \in \mathbb{R}[x_1, .., x_n]$. A *semialgebraic subset* of $\mathbb{R}^n$ is a finite union of basic semialgebraic sets.

We can observe that semialgebraic subsets of $\mathbb{R}^n$ form the smallest family of subsets containing all sets of the form

$$\{\mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) > 0\}, \text{ where } f \in \mathbb{R}[x_1, ..., x_n],$$

that is closed under finite intersections, finite unions, and complements.

Example 8. The semialgebraic subsets of $\mathbb{R}$ are the unions of sets containing finitely many points (the zeros of a polynomial) and open intervals (the solution for $f(x) < 0$ for some polynomial f).

Example 9. We observe also that in Example 2.1 the probability simplex Δ_2 is a semialgebraic set defined by $1 = x + y + z$, $0 \leq x$, $0 \leq y$ and $0 \leq z$.

By proposition 5 we observe that the Zariski closure $\overline{S}$ of a semialgebraic subset S of $\mathbb{R}^m$ is the set of zeros of all polynomials that vanish on S.

Definition 10. The *boundary* ∂S of S is the topological boundary of S (using the standard topology on $\mathbb{R}^m$, not the Zariski topology) inside $\overline{S}$.

2.3 Tensors

Due to their versatility and utility, tensors have various interpretations in different areas of mathematics. For our purposes, tensors are generalizations of matrices to higher dimensions, hence they are n-dimensional arrays of numbers.

Definition 11. Let $V \cong \mathbb{R}^r$ and $W \cong \mathbb{R}^s$ be two vector spaces with $r, s \geq 2$. The *tensor product* $\alpha = v \otimes w$ of the vectors $v = (v_j) \in V$ and $w = (w_j) \in W$ is the array

$$\alpha = [\alpha_{ij}] = [v_i \cdot w_j]$$

of products of the coordinates of v and w. Any tensor product of two vectors is called a *rank-one* matrix. By definition $\otimes$ is a bilinear operation, i.e.,

$$\begin{aligned}(v + v') \otimes w &= v \otimes w + v' \otimes w \\ v \otimes (w + w') &= v \otimes w + v \otimes w' \\ (cv) \otimes w &= v \otimes (cw) = c(v \otimes w)\end{aligned}$$

for all $v, v' \in V$, $w, w' \in W$, and $c \in \mathbb{R}$.

The set of all linear combinations of all tensor products defines the set of tensors.

Definition 12. Let P be a nonnegative real tensor, such that $P = [p_{i_1,i_2,...,i_n}]$ is of format $d_1 \times d_2 \times \cdots \times d_n$ (where format is the number of indices required to uniquely select each component). We say that P has *nonnegative rank* s if it can be written as

$$P = \sum_{t=1}^{s} a_{1t} \otimes a_{2t} \otimes \cdots \otimes a_{nt}$$

where s is minimal and the vectors $a_{it} \in \mathbb{R}^{d_i}$ have nonnegative entries for $i = 1, 2, .., n$, $t = 1, 2, .., s$.

We will be able to express the joint probability distribution of a model with a tensor (making use of the tensor multiplication), and we will be able to identify an n-dimensional probability distribution of format $d_1 \times d_2 \times \cdots \times d_n$ with point in $\Delta_{\prod_{i=1}^{m} d_i - 1}$. We develop this idea more extensively in Chapter 3.

2.4 Latent tree models

2.4.1 Basic graph theory

Definition 13. A *directed graph* G is an ordered pair (V, E), where V is a set whose elements are called *vertices* or *nodes* and E is a set of ordered pairs of vertices called *directed edges*. Given elements $u, v \in V$ and $(u, v) \in E$, we say that u *is a parent of* v and we denote by $Pa(v)$ the set of parents of v. We also say that (u, v) is *adjacent* to v and u. The *degree* of a vertex is the number of adjacent edges. A graph G has a *cycle* if there is a finite sequence of edges $\{a_1, ..., a_n\}$, $n \geq 3$, with no repeated elements such that for any element $a_i = (u, v)$, we have $a_{i+1} = (v, w)$ and such that if $a_1 = (p, q)$ and $a_n = (x, y)$, then $p = y$. A *directed acyclic graph* (DAG) is a directed graph without cycles.

Definition 14. Let $G = (V, E)$ be a directed graph and let $G' = (V, E \cup \{(u, v) | (v, u) \in E\})$. We say that $G = (V, E)$ is connected if for any $u, v \in V$, there is a cycle in G' containing u, v.

Definition 15. We say a directed graph $G = (V, E)$ is a *tree* if it is connected and each node has at most one parent. Let $v \in V(G)$. We say that $v \in V$ is a *root* if it does not have parents and we also say that v is an *inner vertex* if it has degree at least 2. A leaf is a vertex with degree 1.

It can be shown that a directed tree has only one root.

Example 16. Figure 2.2 shows an example of a directed tree with root R, inner vertex A, and leaves X, Y and Z.

2.4.2 Latent tree model

Several statistical models can be represented using a graph whose vertices represent a random variable and every directed edge expresses the dependence structure between two random variables. These types of models are examples of graphical models.

We are very interested in particular parametric models that are also graphical models known as latent tree models. *Latent tree models* are associated to rooted tree-structured graphical models (see for example [MSZTL13],[MTAW00] and [Z16]). Formally, we associate this model with a pair (G, Θ) where G is a DAG tree and Θ is a set of parameters. In a latent

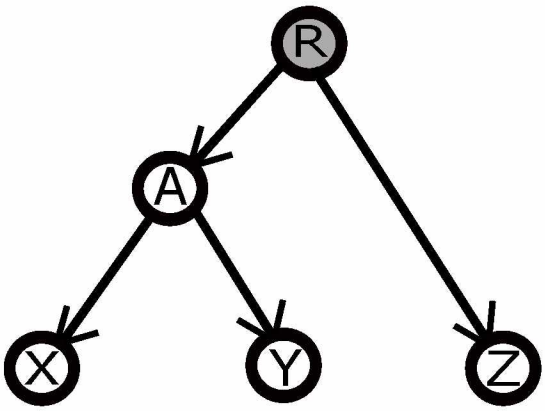

Figure 2.2: Directed tree with root R, inner vertex A, and leaves X, Y and Z.

tree model, each inner node of the tree is assumed to represent a hidden discrete random variable (where a hidden random variable is a variable that is not directly observed) and each leaf vertex is assumed to represent an observable discrete variable. The set of edges $E = \{E_1, ..., E_k\}$ expresses the direct dependences between variables. The set $X = \{X_1, .., X_n\}$ is the set of observed variables and $H = \{H_1, .., H_m\}$ is the set of hidden variables. So the set of nodes $V = \{V_1, ..., V_{m+n}\} = \{H_1, .., H_m, X_1, ..., X_n\}$ represents the $m + n$ hidden and observed variables.

The set Θ consists of Markov matrices $M_{(V_a,V_b)}$ and a row vector π_{V_r}. The Markov matrices have nonnegative entries and rows that sum to 1, one for each $(V_a, V_b) \in E$. As mentioned above, a tree has only one root, so we denote this random variable with V_r. The row vector π_{V_r} correspond to the parentless latent variable V_r. The row vector π_{V_r} specifies the distribution of the random variable V_r, i.e. $\pi_{V_r}(j) = \text{Prob}(V_r = j)$ and each entry of a Markov matrix is $M_{(V_a,V_b)}(i, j) = \text{Prob}(V_b = j|V_a = i)$, i.e. the entries are the transition probabilities of the states of the parent vertex V_a to the child vertex V_b. Denote the set $\{1, 2, ..., k\}$ by $[k]$. Let $V = (V_1, ..., V_{m+n})$ and $\mathbf{j} \in \prod_{i=1}^{n+m}[|V_i|]$, where $|V_i|$ denotes the size of state space of the variable V_i. Then the joint probability distribution of all variables, both observed and latent, is

$$\text{Prob}(V = \mathbf{j}) = \pi_{V_r}(\mathbf{j}_i) \prod_{e \in E} M_e(\mathbf{j}_{V_a}, \mathbf{j}_{V_b}). \tag{2.2}$$

The parametrization map of the joint distribution of all the variables is denoted

$$\phi : \Theta \mapsto \Delta_{(\prod_{i=1}^{n+m} |V_i|)-1}.$$

Since the probability distribution of the model with hidden variables is obtained from the fully observed model, we can obtain the probability distribution P of the observable variables by marginalizing over the hidden variables, where the parametrization map is

$$\psi = \delta \circ \phi : \Theta \mapsto \Delta_{(\prod_{i=1}^{n} |X_i|)-1},$$

with δ denoting the appropriate map marginalizing over the hidden variables. For any vector $\mathbf{x} = (x_1, x_2, ..., x_n) \in \mathbb{R}^n$, let $\mathbf{x}_i$ denote the i-th entry of $\mathbf{x}$, i.e. $\mathbf{x}_i = x_i$. If $\mathbf{t} = (t_{m+1}, t_{m+2}, ..., t_n) \in \prod_{j=m+1}^{n}[|V_j|]$, then for any $\mathbf{s} = (s_1, s_2, .., s_m) \in \prod_{j=1}^{m}[|V_j|]$ we denote $(\mathbf{s}, \mathbf{t}) = (s_1, s_2, .., s_m, t_{m+1}, t_{m+2}, ..t_n)$. Thus

$$\text{Prob}(\mathbf{t}) = \sum_{\mathbf{s} \in \prod_{j=1}^{m}[|V_j|]} \pi_{V_r}((\mathbf{s}, \mathbf{t})_i) \prod_{e \in E} M_e((\mathbf{s}, \mathbf{t})_{V_a}, (\mathbf{s}, \mathbf{t})_{V_b}). \tag{2.3}$$

Then $\psi(\Theta)$ is the collection of distributions on n observable variables that arise from the latent class model.

At this point an example will come in handy, so we will see a particular one which we are very interested in, known as the 3-leaf model. Since we will look for a lot of properties of this model we will dedicate a whole chapter to it.

Chapter 3

The 3-leaf model

Let $\mathcal{M}$ be a latent tree model associated with (S_3, Θ) with tree S_3 shown in Figure 3.1, where H is a hidden variable with state space of size k and X_1, X_2, X_3 are observable variables with state spaces of sizes k_1, k_2, k_3 respectively. The set of parameters Θ of $\mathcal{M}$ is the set of transition matrices: M_1 of format $k \times k_1$ corresponding to the variable X_1, M_2 of format $k \times k_2$ corresponding to the variable X_2, M_3 of format $k \times k_3$ corresponding to the variable X_3, and π of format $1 \times k$ corresponding to the variable H. We observe from (2.2) that the joint probability of all the variables, both hidden and observable, is

$$P(H = h, X_1 = x_1, X_2 = x_2, X_3 = x_3) = \pi(h) \cdot M_1(h, x_1) \cdot M_2(h, x_2) \cdot M_3(h, x_3).$$

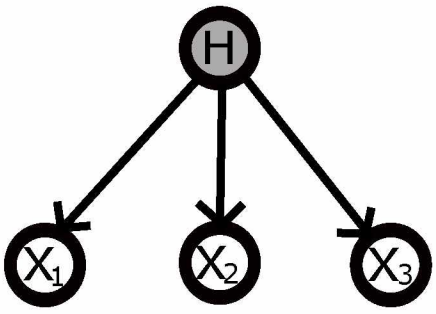

Figure 3.1: S_3: The directed tree associated to the model $\mathcal{M}$

By marginalizing over the latent variable, we obtain the distribution of all observable variables for the model $\mathcal{M}$:

$$P(X_1 = x_1, X_2 = x_2, X_3 = x_3) = \sum_{i=1}^{k} \pi(i) \cdot M_1(i, x_1) \cdot M_2(i, x_2) \cdot M_3(i, x_3). \tag{3.1}$$

By conditional independence of the variables X_1, X_2 and X_3 given the variable H we observe that we can express the distribution of the observable variables as

$$P(X_1, X_2, X_3) = \sum_{i=1}^{k} \pi(i)(M_1(i,:) \otimes M_2(i,:) \otimes M_3(i,:)), \tag{3.2}$$

where $M_j(i,:)$ is the i-th row of the matrix M_j, $j = 1, 2, 3$. In particular, this expression allow us to see the parametrization map of $\mathcal{M}$:

$$\psi : \Theta = \{\{\pi, M_1, M_2, M_3\}\} \mapsto \Delta_{q-1} \tag{3.3}$$

where $q = \prod_{i=1}^{3} k_i$.

We observe that the Markov matrix M_i has format $k \times k_i$, and since in each row the entries sum to 1, the last column is determined by the first $k_i - 1$ columns for $i = 1, 2, 3$. Also we note that since the vector π has format $1 \times k$ and the entries sum to 1, the last entry is determined by the first $k - 1$ entries. Thus

$$\dim(\Theta) = k - 1 + k(k_1 - 1) + k(k_2 - 1) + k(k_3 - 1). \tag{3.4}$$

The coordinate functions of ψ are polynomials. This property is crucial to us.

Example 17. Consider the model $\mathcal{M}$ when $k = k_1 = k_2 = k_3 = 2$, denoted $\mathcal{M}_2$, also known as the binary 3-leaf model. Let

$$\theta = \left\{\pi = (0.7, 0.3), M_1 = \begin{pmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{pmatrix}, M_2 = \begin{pmatrix} 0.6 & 0.4 \\ 0.1 & 0.9 \end{pmatrix}, M_3 = \begin{pmatrix} 0.7 & 0.3 \\ 0.25 & 0.75 \end{pmatrix}\right\}.$$

Then the distribution $P = P(\theta)$ of all observable variables is

$$P = 0.7 \cdot (M_1(1,:) \otimes M_2(1,:) \otimes M_3(1,:)) + 0.3 \cdot (M_1(2,:) \otimes M_2(2,:) \otimes M_3(2,:)).$$

Thus

$$P = \left[\begin{array}{cc|cc} 0.2382 & 0.1838 & 0.1098 & 0.1482 \\ 0.0633 & 0.0797 & 0.0387 & 0.1383 \end{array}\right].$$

We observe from (3.4) that $\dim(\Theta) = 7$.

Definition 18. Consider the map ψ as defined in (3.3). We refer to this probabilistic model as having *stochastic parameters* and a *stochastic parametrization map* ψ_{st}. We can define a new map by dropping the nonnegativity assumptions of the parameters but retaining the condition that the rows of the Markov matrices and the entries of π sum to 1. We consider *complex parameters* and the *complex parametrization map* $\psi_{\mathbb{C}}$ when the parameter space has complex entries, as well as the *real parameters* and the *real parametrization map* $\psi_{\mathbb{R}}$ when the parameter space has real entries [ART14].

Remark 1. The image of complex, real or stochastic parameters under ψ is a 3-dimensional $k_1 \times k_2 \times k_3$ tensor, whose $\prod_{i=1}^{3} k_i$ entries sum to 1.

Example 19. In the model $\mathcal{M}_2$, the parameters

$$\theta_1 = \Big\{\pi = (1.1835, -0.1835), M_1 = \begin{pmatrix} 1.5109 & -0.5109 \\ 0.6972 & 0.3028 \end{pmatrix}, M_2 = \begin{pmatrix} 0.2666 & 0.7334 \\ 0.1276 & 0.8724 \end{pmatrix},$$
$$M_3 = \begin{pmatrix} 1.5888 & -0.5888 \\ 0.7708 & 0.2292 \end{pmatrix} \Big\}$$

satisfies

$$\psi_{\mathbb{R}}(\theta_1) = \left[\begin{array}{cc|cc} 0.1134 & 0.0821 & 0.0713 & 0.2811 \\ 0.0927 & 0.3325 & 0.0149 & 0.0121 \end{array}\right] \in \Delta_7$$

and the parameters

$$\theta_2 = \Big\{\pi = (0.5 + 1.8552i, 0.5 - 1.8552i), M_1 = \begin{pmatrix} 0.1084 + 0.0827i & 0.8916 - 0.0827i \\ 0.1084 - 0.0827i & 00.8916 + 0.0827i \end{pmatrix},$$
$$M_2 = \begin{pmatrix} -0.0387 - 0.0803i & 1.0387 + 0.0803i \\ -0.0387 + 0.0803i & 1.0387 - 0.0803i \end{pmatrix}, M_3 = \begin{pmatrix} 0.3207 + 0.0474i & 0.6793 + 0.0474i \\ 0.3207 - 0.0474i & 0.6793 - 0.0474i \end{pmatrix} \Big\}$$

satisfies

$$\psi_{\mathbb{C}}(\theta_2) = \left[\begin{array}{cc|cc} 0.0009 & 0.1474 & 0.0087 & 0.2582 \\ 0.0716 & 0.2768 & 0.1781 & 0.0583 \end{array}\right] \in \Delta_7.$$

Example 20. In the model $\mathcal{M}_2$, the parameters

$$\theta_1 = \Big\{\pi = (0.7, 0.3), M_1 = \begin{pmatrix} 1.2 & -0.2 \\ 0.9 & 0.1 \end{pmatrix}, M_2 = \begin{pmatrix} 0.8 & 0.2 \\ 1.2 & -0.2 \end{pmatrix},$$
$$M_3 = \begin{pmatrix} 3 & -2 \\ 0.2 & 0.8 \end{pmatrix} \Big\}$$

satisfies

$$\psi_{\mathbb{R}}(\theta_1) = \left[\begin{array}{cc|cc} 2.0808 & 0.4932 & -1.0848 & -0.3792 \\ -0.3288 & -0.0852 & 0.2528 & 0.0512 \end{array}\right] \notin \Delta_7$$

and the parameters

$$\theta_2 = \Big\{\pi = (0.7, 0, 3), M_1 = \begin{pmatrix} 0.7 + 0.7i & 0.3 - 0.7i \\ 0.4 + 0.4i & 0.6 - 0.4i \end{pmatrix},$$
$$M_2 = \begin{pmatrix} 0.6 + i & 0.4 - i \\ 0.1 + 0.3i & 0.9 - 0.3i \end{pmatrix}, M_3 = \begin{pmatrix} 1.2 + 0.2i & -0.2 - 0.2i \\ 0.3 + 0.9i & 0.7 - 0.9i \end{pmatrix} \Big\}$$

satisfies

$$\psi_{\mathbb{C}}(\theta_2) = \left[\begin{array}{cc|cc} -0.2856 + 0.3696i & 0.3276 + 0.2604i & 0.1456 + 0.0784i & 0.3024 - 0.2184i \\ 0.2736 + 0.1524i & 0.2544 - 0.0924i & 0.1164 - 0.0904i & -0.1344 - 0.4596i \end{array}\right] \notin \Delta_7.$$

Trivially $\mathrm{Im}(\psi_{st}(\Theta)) \subseteq \mathrm{Im}(\psi_{\mathbb{R}}(\Theta)) \subseteq \mathrm{Im}(\psi_{\mathbb{C}}(\Theta))$, but we wonder how these sets differ. Under the Zariski topology for $\mathbb{C}$, it is a known that

$$\mathrm{Im}(\psi_{st}(\Theta)) \subsetneq \Delta_{q-1} \subsetneq \overline{\mathrm{Im}(\psi_{\mathbb{R}}(\Theta))} \subseteq \overline{\mathrm{Im}(\psi_{\mathbb{C}}(\Theta))} = V.$$

In particular, by Example 19, we observe that Δ_{q-1} contains points of $\mathrm{Im}(\psi_{\mathbb{C}}(\Theta))\backslash\mathrm{Im}(\psi_{st}(\Theta))$ and points of $\mathrm{Im}(\psi_{\mathbb{R}}(\Theta)) \setminus \mathrm{Im}(\psi_{st}(\Theta))$ (it also contains other points). By Example 20 we observe that $\Delta_{q-1} \neq \mathrm{Im}(\psi_{\mathbb{R}}(\Theta))$ and that $\Delta_{q-1} \neq \mathrm{Im}(\psi_{\mathbb{C}}(\Theta))$. We note that V is defined by a single equation $\sum p_{ijk} = 1$.

Denote by $\mathcal{M}_3$ the model $\mathcal{M}$ when $k = k_1 = k_2 = 3$ and $k_3 = 2$.

Remark 2. Since the coordinate functions of the parametrization map of the models $\mathcal{M}_2$ and $\mathcal{M}_3$ are polynomials, we can talk about the Jacobian of the map. We compute the rank of the Jacobian at a generic point of the interior of the parameter space of $\mathcal{M}_2$ and the rank of the Jacobian at a generic point of the interior of the parameter space of $\mathcal{M}_3$. For the model $\mathcal{M}_2$, the rank of the Jacobian at a generic point in the interior of the parameter space is 7. For the model $\mathcal{M}_3$, the rank of the Jacobian at a generic point in the interior of the parameter space is 17. Therefore $\dim(\mathrm{Im}(\psi_2)) = 7$ and $\dim(\mathrm{Im}(\psi_3)) = 17$, where ψ_i denotes the parametrization of the model $\mathcal{M}_i$.

We are now ready to state a theorem of crucial importance for us.

Theorem 21. *The intersection of the probability simplex and the Zariski closure of the image of ψ_2 is the whole probability simplex Δ_7, i.e. $\overline{Im(\psi_2)} \cap \Delta_7 = \Delta_7$ and the intersection of the probability simplex and with the Zariski closure of the image of ψ_3 is the whole probability simplex Δ_{17}, i.e. $\overline{Im(\psi_3)} \cap \Delta_{17} = \Delta_{17}$.*

This is because $\dim(\mathrm{Im}(\psi_2)) = 7$ and $\dim(\mathrm{Im}(\psi_3)) = 17$, which are full dimensional open subsets of Δ_7 and Δ_{17} respectively.

For any latent tree model $\mathcal{M}$, the property that $\overline{\mathrm{Im}(\psi)} \cap \Delta_{(\prod_{i\in[3]} n_i)-1} = \Delta_{(\prod_{i\in[3]} n_i)-1}$ is false in general. That is why we are very interested in the models $\mathcal{M}_2$ and $\mathcal{M}_3$. For example, denote by $\mathcal{M}^*$ the model $\mathcal{M}$ where $k = k_1 = k_2 = k_3 = 3$. The problem with $\mathcal{M}^*$ is that the parameter space has dimension 20 and the simplex where the image of the parametric map ψ_* lies has dimension 26. Thus $\overline{\mathrm{Im}(\psi_*)} \subsetneq \Delta_{26}$ since $\dim(\overline{\mathrm{Im}(\psi_*)}) < \dim(\Delta_{26})$.

3.0.3 Parameter identifiability

A model is *strictly identifiable* if different values of parameters generate different probability distributions (in other words, if the parametrization map ψ is one-to-one). In general, strict identifiability is a very strong property for a latent class model, so it is too much to ask for strict identifiability. For the model $\mathcal{M}$, let $\{\pi, M_1, M_2, M_3\} = \theta \in \Theta$, then by label swapping (see example 22) the states of π we will obtain $k! - 1$ other elements $\theta_j \in \Theta$ such that $\psi(\theta_j) = \psi(\theta)$. With this we can see that our parametrization map is not one-to-one, but what can we expect for some source of identifiability? It is possible to prove that for any generic parameters $\theta_1, \theta_2 \in \Theta$ such that $\psi(\theta_1) = \psi(\theta_2)$, θ_1 and θ_2 differ only up to label swapping, which is equivalent to say that ψ is generically $k!$-to-one (see Theorem 23). We say that our model is *identifiable up to label swapping*. For example, generically ψ_2 is generically 2-to-one and ψ_3 is generically 6-to-one

Example 22. Consider the model $\mathcal{M}_2$. Let

$$\theta^* = \left\{\pi = (0.3, 0.7), M_1 = \begin{pmatrix} 0.4 & 0.6 \\ 0.8 & 0.2 \end{pmatrix}, M_2 = \begin{pmatrix} 0.1 & 0.9 \\ 0.6 & 0.4 \end{pmatrix}, M_3 = \begin{pmatrix} 0.25 & 0.75 \\ 0.7 & 0.3 \end{pmatrix}\right\}$$

Then the distribution P of all observable variables is

$$\psi(\theta^*) = 0.7 \cdot (M_1(1,:) \otimes M_2(1,:) \otimes M_3(1,:)) + 0.3 \cdot (M_1(2,:) \otimes M_2(2,:) \otimes M_3(2,:))$$

Thus

$$\psi(\theta^*) = \left[\begin{array}{cc|cc} 0.2382 & 0.1838 & 0.1098 & 0.1482 \\ 0.0633 & 0.0797 & 0.0387 & 0.1383 \end{array}\right].$$

We observe that in Example 17 we obtain the same tensor with different parameters. Note that the parameter θ^* differs from the parameter in Example 17 just by label swapping. The Markov matrices have the same rows but in different order; the vector π is obtained by switching the values.

The next natural question is: When does a probability distribution P arise from stochastic parameters? This question is answered in various papers; see for example [ARSV14]. The next theorem is an specific case for our models $\mathcal{M}_2$ and $\mathcal{M}_3$.

Theorem 23. *Consider the models $\mathcal{M}_2$ and $\mathcal{M}_3$. Then generic parameters of these models are identifiable up to label swapping and there exists an algebraic procedure for the determinations of the parameters from the joint probability distribution $P(X_1, X_2, X_3)$.*

Using the steps of the proof in [ARSV14] we programmed the functions `params2x2x2.m` and `params3x3x2.m` in Matlab (see the Appendix). These functions have as input a $2\times2\times2$ and $3\times3\times2$ probability distribution (nonnegative tensor whose entries sum to 1), respectively, and have as output the parameters M_1, M_2, M_3 and π up to label swapping. This is assuming that the code returns something. There are some cases, like when $\pi = (1, 0)$, where the code does not work properly. The set of these cases have measure zero in $[0, 1]^{\dim(\Theta)}$.

Example 24. Let us go through the code with a particular example. Let

$$P = \left[\begin{array}{cc|cc} 0.2382 & 0.1838 & 0.1098 & 0.1482 \\ 0.0633 & 0.0797 & 0.0387 & 0.1383 \end{array}\right].$$

Let P_3 be the marginalization of P over the observable variable X_3. It is possible to compute

$$P_3 = M_1^T \mathrm{diag}(\pi) M_2 = \begin{pmatrix} 0.3480 & 0.3320 \\ 0.1020 & 0.2180 \end{pmatrix}.$$

Let $P_{..i}$ be the slice of P with third index fixed at i;

$$P_{..1} = M_1^T \mathrm{diag}(\pi)\mathrm{diag}(M_3(:,1)) M_2 = \begin{pmatrix} 0.2382 & 0.1838 \\ 0.0633 & 0.0797 \end{pmatrix}$$

and

$$P_{..2} = M_1^T \mathrm{diag}(\pi)\mathrm{diag}(M_3(:,2)) M_2 = \begin{pmatrix} 0.1098 & 0.1482 \\ 0.0387 & 0.1383 \end{pmatrix}.$$

Assuming that P_3 is invertible (this assumption might seem strong, but the set of tensors for which P_3 is singular has measure zero), we compute

$$P_3^{-1} P_{..1} = M_2^{-1} \mathrm{diag}(M_3(:,1)) M_2 = \begin{pmatrix} 0.7360 & 0.3240 \\ -0.0540 & 0.2140 \end{pmatrix}. \tag{3.5}$$

Thus the columns of M_3 are determined by the eigenvalues of $P_3^{-1}P_{\cdot 1}$ and $P_3^{-1}P_{\cdot 2}$ (this is because $P_3^{-1}P_{\cdot 1}$ is a diagonalizable matrix as shown on (3.5)). These are

$$\begin{pmatrix} 0.70 \\ 0.25 \end{pmatrix} \text{ and } \begin{pmatrix} 0.3 \\ 0.75 \end{pmatrix} \text{ respectively.}$$

Thus $M_3 = \begin{pmatrix} 0.7 & 0.3 \\ 0.25 & 0.75 \end{pmatrix}$. To obtain the rows of M_2 first we compute the left eigenvectors of $P_3^{-1}P_{\cdot 1}$ and then we normalize each row (assuming the sum of the row entries is not zero we divide each entry by it) so the entries sum to 1. Thus the left eigenvectors of $P_3^{-1}P_{\cdot 1}$ are $\begin{pmatrix} 0.8321 & 0.1104 \\ 0.5547 & 0.9939 \end{pmatrix}$ and by normalizing we obtain

$$M_2 = \begin{pmatrix} 0.6 & 0.4 \\ 0.1 & 0.9 \end{pmatrix}.$$

Similarly we compute M_1 with $P_{\cdot i}P_3^{-1}$ for $i = 1, 2$. So we obtain $M_1 = \begin{pmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{pmatrix}$. Now that we computed M_1 and M_2, π is the diagonal of the matrix $M_1^{-T}P_3M_2^{-1}$. Note that

$$M_1^{-T}P_3M_2^{-1} = \begin{pmatrix} 0.7 & 0.0 \\ 0.0 & 0.3 \end{pmatrix}.$$

Thus $\pi = (0.7, 0.3)$, which agrees with Example 17.

3.1 Volume of the model in the probability simplex

One of the main interests of this work is determine the volume of $\mathcal{M}_2$ and $\mathcal{M}_3$ in their respective probability simplices. We are interested in the volume since by Theorem 21, we know that $\overline{\text{Im}(\psi_3)} \cap \Delta_{17} = \Delta_{17}$ and $\overline{\text{Im}(\psi_2)} \cap \Delta_7 = \Delta_7$, thus $\text{Im}(\psi_3)$ is Zariski dense in Δ_{17} and $\text{Im}(\psi_2)$ is Zariski dense in Δ_7 .

3.1.1 The volume of $\mathcal{M}_2$ in Δ_7

Since the model $\mathcal{M}_2$ satisfies $\overline{\mathrm{Im}(\psi_2)} \cap \Delta_7 = \Delta_7$, we consider the volume of $\mathcal{M}_2$ in Δ_7.

In order to estimate the volume of $\mathcal{M}_2$ in Δ_7, we programmed the function `Sto2x2x2.m` in Matlab (see the Appendix). This function takes a sample of size n from the probability simplex. The sample $\{x_1, x_2, ..., x_n\}$ with $x_i \in \Delta_7$ is chosen using a Dirichlet distribution [KBJ00].

Note that x_i can be rearranged in tensor form for $i = 1, 2, ..., n$. We apply `params2x2x2.m` to each x_i in the sample, and that returns the parameter π, M_1, M_2 and M_3 for x_i. We then verify the entries of each parameter to determine if they are stochastic, real non-stochastic or complex non-real. In `params2x2x2.m` we assume that P_3 is invertible, which is equivalent to the assumption that π has no zero entries and M_1 and M_3 are invertible. After several runs of the program we obtain that for a sample of size $n = 10^5$, the percentage of distributions that arise from stochastic parameters is 8.3% (this agrees with [ZS11]), the percentage of distributions that arise from real, non-stochastic parameters is 81.5% and the percentage of distributions that arise from complex, non-real parameters is 10.2%.

3.1.2 The volume of $\mathcal{M}_3$ in Δ_{17}

Analogously to the previous section, we can take a sample $x_1, ..., x_n$ of some distributions in Δ_{17} and apply `params3x3x2.m` to each x_i. This function returns $\{\pi, M_1, M_2, M_3\}$ that give rise to x_i. Again we verify the entries of each element in the parameter to determine if they are stochastic, real non-stochastic or complex non-real.

After several runs of the program, for a sample of size $n = 10^5$ we obtain the following estimations:

Estimation 25. *The rough percentage of distributions that arise from stochastic parameters is* 0.015%*, the rough percentage of distributions that arise from real, non-stochastic parameters is* 64.742% *and the rough percentage of distributions that arise from complex, non-real parameters is* 35.206%*.*

These percentages do not sum exactly to 1 due to numerical error. We actually ran tests with higher values of n but the change was not very significant.

Even though they were not used in the code, results like the next proposition facilitate the classification of parameters; the next proposition can also give us the main idea of the proof of Theorem 23.

Proposition 26. *Let $P \in \mathbb{R}^{3\times3\times2}$ be a distribution (tensor) of nonnegative rank 3. If $M_3 \in \mathbb{R}^{3\times2}$, then $M_1, M_2 \in \mathbb{R}^{3\times3}$.*

Proof. Let $P_{..+}(i,j) = \sum_{k=1}^{2} P(i,j,k)$ (i.e. summing over the third index; see P_3 in Example 24). We observe that

$$P_{..+} = M_1^T \mathrm{diag}(\pi) M_2.$$

Let $P_{..k}(i,j) = P(i,j,k)$. Then

$$P_{..k} = M_1^T \mathrm{diag}(\pi)\mathrm{diag}(M_3(:,k)) M_2.$$

Assuming that M_2, M_1 are non-singular and π has non zero entries, then $P_{..+}$ is invertible and we see

$$P_{..+}^{-1} P_{..k} = M_2^{-1}\mathrm{diag}(M_3(:,k)) M_2$$

Since all entries of M_3 are real, all the eigenvalues of $P_{..+}^{-1}P_{..k}$ are real. Let λ be an eigenvalue of $P_{..+}^{-1}P_{..k}$. Thus a left eigenvector v associated with λ of $P_{..+}^{-1}P_{..k}$ satisfies

$$\begin{aligned} v P_{..+}^{-1} P_{..k} &= \lambda v, \\ (P_{..+}^{-1} P_{..k})^T v^T &= \lambda v^T, \\ (P_{..+}^{-1} P_{..k} - \lambda I)^T v^T &= 0, \\ ((P_{..+}^{-1} P_{..k})^T - \lambda I) v^T &= 0. \end{aligned}$$

Since $(P_{..+}^{-1}P_{..k})^T - \lambda I$ is real, then v^T must be real. Therefore the entries of M_2 are real. Analogously, since

$$P_{..k} P_{..+}^{-1} = M_1^{-1}\mathrm{diag}(M_3(:,k)) M_1,$$

the entries of M_1 are real. □

Now that we have established the volume of both models in their correspondent probability simplices, in the next chapter we will explore some of their geometric properties.

Chapter 4

The algebraic boundary of $\mathcal{M}$

4.1 Algebraic boundary

Let $P = [p_{i_1,i_2,..,i_n}]$ be a real n-dimensional tensor of format $d_1 \times d_2 \times \cdots \times d_n$.

Definition 27. We define the *algebraic boundary* of a semialgebraic subset $S \subset \mathbb{R}^d$ to be the Zariski closure $\overline{\partial S}$ of its topological boundary. This is equivalent to the smallest algebraic variety containing its boundary in the Euclidean topology.

Definition 28. A *slice* of a tensor P is a subtensor of some format $d_1 \times d_2 \times \cdots d_{s-1} \times 1 \times d_{s+1} \times \cdots \times d_n$.

For example, note that in Example 24 the matrix defined as $P_{..i}$ is a slice of P of format $2 \times 2 \times 1$.

The next theorem is proved in [ARSZ15] and concerns the algebraic boundary of the model $\mathcal{M}$.

Theorem 29. *[ARSZ15] The algebraic boundary of $\mathcal{M}_d$ has $\sum_{i=1}^{3} k_i$ irreducible components for $d = 2, 3$. In particular, the irreducible components of $\mathcal{M}_2$ are given by slices having rank ≤ 1.*

This implies that the algebraic boundary of the model $\mathcal{M}_2$ has 6 irreducible components and the algebraic boundary of the model $\mathcal{M}_3$ has 8. Let us discuss why.

We will see that each irreducible component corresponds to having a zero in a row of any Markov matrix M_i. Let

$$\pi = (\pi_1, \pi_2),\ M_1 = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix},\ M_2 = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix},\ \text{and } M_3 = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$$

be the parameters of $p \in \mathcal{M}_2$.

Note that our parameter space Θ can be identified with a subset of $[0,1]^{\dim(\Theta)}$. Thus, in an informal way for $\mathcal{M}_2$, we can think that if a parameter θ has at least one zero, then θ lives in the boundary of $[0,1]^7$. Thus $\psi(\theta)$ might live in the boundary of $\mathcal{M}$ (ψ is a 2-to-1 polynomial map) and therefore might live in $\overline{\partial\mathcal{M}_2}$. Let $\{a_{11}=0\}$ be the set of all parameters with $a_{11}=0$. For example, in any neighbourhood $\mathcal{U} \subset \mathbb{R}^7$ of a point in $\{a_{11}=0\}$ (or $\{a_{12}=0\}$), there are parameters where $a_{11}<0$ that are not mapped into the interior of $\mathcal{M}$. In particular we observe that since $\{a_{11}=0\}$ lives in a facet of the cube $[0,1]^7$, then $\dim(\psi(\{a_{11}=0\})) < \dim(\Delta_7)$. Doing a similar computation as the one mentioned in Remark 2, we obtain that the rank of the Jacobian at a generic point in $\{a_{11}=0\} \cap \mathcal{U}$ is 6. Thus the dimension of the irreducible components of $\partial\mathcal{M}_2$ is equal to 6.

Now we will talk about how we have completely described the irreducible components of the algebraic boundary of $\mathcal{M}_2$. Let θ be a parameter such that the entries of the Markov matrices are all positive and that is mapped to $\partial\mathcal{M}_2$ (such points are called singular points). In [ARSZ15] it is proven that θ either has a zero in π or a Markov matrix is singular. It is also proven in [ARSZ15] that there exists $\theta^* \in [0,1]^7$ with a zero entry in a Markov matrix such that $\psi(\theta^*)=\psi(\theta)$. This implies that the algebraic boundary of $\mathcal{M}_2$ is completely described by the set of parameters with a zero entry in a Markov matrix.

Briefly we show why there exists $\theta^* \in [0,1]^7$ with a zero entry in a Markov matrix such that $\psi(\theta^*)=\psi(\theta)$. Suppose first that $\theta=\{\pi, M_1, M_2, M_3\}$ has a zero on π. Without loss of generality suppose that $\pi=(0,1)$. Thus, we can define $\theta^*=\{\pi, M_1^*, M_2, M_3\}$ where M_1^* is M_1 but with first row $(0,1)$. Note that M_1 has a zero entry and $\psi(\theta^*)=\psi(\theta)$. Now suppose that a Markov matrix of θ is singular. Without loss of generality suppose that M_1 is singular. Let $\theta^*=\{\pi^*, M_1^*, M_2^*, M_3^*\}$ be such that $\pi^*=\pi M_1$, $M_1^*=M_1$, M_2^* be the identity matrix and $M_3^*=\text{diag}(\pi^*)^{-1}M_2^T\text{diag}(\pi)M_3$. Note that M_2^* has a zero entry and by doing the calculations we can observe that $\psi(\theta)=\psi(\theta^*)$ [ARSZ15].

It can be shown that $\dim(\psi(\{\pi_1=0\}))=3$ by obtaining the rank of the Jacobian at a generic point in $\{\pi_1=0\}$.

We can now proceed to characterize the irreducible components.

4.2 Boundary strata of $\mathcal{M}_2$

As mentioned above there are 6 irreducible components on the algebraic boundary of $\mathcal{M}_2$. The irreducible components of the algebraic boundary $\overline{\partial\mathcal{M}_2}$ are:

a) Two 6 dimensional components F6c1k, $k = 1, 2$, are given by $\overline{\psi\{c_{1k} = 0\}} = \overline{\psi\{c_{2k} = 0\}}$. Note that this can be identified with the set of tensors such that the determinants of P_1 and P_2 are zero, where P_1 and P_2 are the 2×2-slice $P_k = [p_{**k}]$. This follows from the observation that the determinants of P_1 and P_2 are

$$\det(P_2) = p_{112}p_{222} - p_{122}p_{212} = c_{12}c_{22}\pi_1\pi_2 \det(M_1)\det(M_2) = 0,$$

$$\det(P_1) = p_{221}p_{221} - p_{121}p_{211} = c_{11}c_{21}\pi_1\pi_2 \det(M_1)\det(M_2) = 0.$$

b) Two 6 dimensional components F6a1i, $i = 1, 2$, are given by $\overline{\psi\{a_{1k} = 0\}} = \overline{\psi\{a_{2k} = 0\}}$. Note that this can be identified with the set of tensors such that the determinants of the 2×2-slice P_{a_1} and P_{a_2} are zero, where $P_{a_2} = [p_{2,i,j}]$ and $P_{a_1} = [p_{1,i,j}]$. This follows from the observation that the determinants of P_{a_2} and P_{a_1} are

$$\det(P_{a_2}) = p_{211}p_{222} - p_{212}p_{221} = a_{12}a_{22}\pi_1\pi_2 \det(M_2)\det(M_3) = 0,$$

$$\det(P_{a_1}) = p_{111}p_{122} - p_{112}p_{121} = a_{11}a_{21}\pi_1\pi_2 \det(M_2)\det(M_3) = 0.$$

c) Two 6 dimensional components F6b1i, $i = 1, 2$, are given by $\overline{\psi\{b_{1k} = 0\}} = \overline{\psi\{b_{2k} = 0\}}$. Note that this can be identified with the set of tensors such that the determinants of the 2×2-slice P_{b_2} and P_{b_1} are zero, where $P_{b_2} = [p_{i,2,j}]$ and $P_{b_1} = [p_{i,1,j}]$. This follows from the observation that the determinants of P_{b_2} and P_{b_1} are

$$\det(P_{b_2}) = p_{121}p_{222} - p_{122}p_{221} = b_{12}b_{22}\pi_1\pi_2 \det(M_1)\det(M_3) = 0,$$

$$\det(P_{b_1}) = p_{111}p_{212} - p_{112}p_{211} = b_{11}b_{21}\pi_1\pi_2 \det(M_1)\det(M_3) = 0.$$

Thus the algebraic boundary of $\mathcal{M}_2$ is the union of its irreducible components as follows:

$$\begin{aligned}\{p_{112}p_{222} = p_{122}p_{212}\} \cup \{p_{121}p_{222} = p_{122}p_{221}\} \cup \{p_{211}p_{222} = p_{212}p_{221}\} \\ \cup \{p_{111}p_{122} = p_{112}p_{121}\} \cup \{p_{111}p_{212} = p_{112}p_{211}\} \\ \cup \{p_{221}p_{222} = p_{121}p_{211}\}.\end{aligned}$$

As mentioned above, the dimension of each component is 6. Table 4.1 shows a classification of the irreducible components of the boundary of $\mathcal{M}_2$ up to label swapping and the interior according to the zeros in the parameters.

Table 4.1: Boundary strata of $\mathcal{M}_2$ distributions generically of rank 2

Name	#	Long name	Representative Parameters	Comments
F7	1	F7	No zeros	Interior of $\mathcal{M}$
There are **6** irreducible components in the algebraic boundary $\overline{\partial\mathcal{M}}$				
F6	6	F6a11 F6a12 F6b11 F6b12 F6c11 F6c12	$M_i = \begin{pmatrix} 0 & 1 \\ * & * \end{pmatrix}$	One zero in a matrix
Fπ	1			Zero in π.
Other				More than 1 zero in the parameters (no zero in π)

4.3 Boundary strata of $\mathcal{M}_3$

The details to determine the algebraic boundary of M_3 follow the same spirit as the case $\mathcal{M}_2$ and are found in [ARSZ15]. Let

$$\pi = (\pi_1, \pi_2, \pi_3),\ M_1 = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix},\ M_2 = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix},\ \text{and } M_3 = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ c_{31} & c_{32} \end{pmatrix}$$

be the parameters of $p \in \mathcal{M}_3$.

The irreducible components of the algebraic boundary $\overline{\partial\mathcal{M}_3}$ are:

a) The two components F16ck, $k = 1, 2$, are given by the determinants of P_1 and P_2. This are $\overline{\varphi\{c_{1k} = 0\}} = \overline{\varphi\{c_{2k} = 0\}} = \overline{\varphi\{c_{3k} = 0\}}$, given by the 3×3-slice $P_k = \{p_{**k}\}$ having rank ≤ 2. This corresponds to a zero in M_3.

b) The three components F16a1i, $i = 1, 2, 3$, are given by the 3×3 matrix $P_1 \cdot (P_2)^{-1} = M_1^T \Lambda_1 M_1^{-T}$ having an eigenvector with zero i-th coordinate (where $\Lambda_1 = \text{diag}(M_3(:, 1) \cdot \text{diag}(M_3(:, 2))^{-1})$. This corresponds to a zero in M_1.

c) The three components F16b1j, $j = 1, 2, 3$, are given by the 3×3 matrix $P_1^T \cdot (P_2)^{-T} = M_2^T \Lambda_2 M_2^{-T}$ having an eigenvector with zero j-th coordinate (where $\Lambda_2 = \text{diag}(M_3(:, 1)^T \cdot \text{diag}(M_3(:, 2))^{-T})$. This corresponds to a zero in M_2.

Table 4.2 shows a classification of the irreducible components of the boundary of $\mathcal{M}_3$ and the interior according to the zeros in the parameters.

Now that we have talked about some structure of our model $\mathcal{M}_2$ and $\mathcal{M}_3$, we are going to make use of it with a well known tool in statistics called the EM algorithm. In the next chapter we discuss this tool and our interest in it.

Table 4.2: Boundary strata of $\mathcal{M}_3$ distributions generically of rank 3

Name	#	Long name	Representative Parameters	Comments
F17	1	F17	No zeros	Interior of $\mathcal{M}$
There are **8** irreducible components in the algebraic boundary $\overline{\partial\mathcal{M}}$				
F16	3	F16a11 F16a12 F16a13	$\begin{pmatrix} 0 & a_{12} & a_{13} \\ * & * & * \\ * & * & * \end{pmatrix}$	An eigenvector of $P_1 \cdot (P_2)^{-1}$ with zero i-th coordinate, for $i = 1, 2, 3$.
	3	F16b11 F16b12 F16b13	$\begin{pmatrix} 0 & b_{12} & b_{13} \\ * & * & * \\ * & * & * \end{pmatrix}$	An eigenvector of $P_1^T \cdot (P_2)^{-T}$ with zero j-th coordinate, for $j = 1, 2, 3$.
	2	F16c11 F16c12	$\begin{pmatrix} 0 & c_{12} \\ * & * \\ * & * \end{pmatrix}$	The determinants of P_1 and P_2
Fπ	1			Zero in π.
Other				More than 1 zero in the parameters (no zero in π).

Chapter 5

The EM algorithm

5.1 Maximum likelihood estimator

Definition 30. Let $X_1, ..., X_n$ be independent and identically distributed discrete random variables with probability mass function (pmf) $f(x;\theta)$. The *likelihood function* is the function is defined by

$$\mathcal{L}(\theta) = \prod_{i=1}^{n} f(X_i;\theta).$$

If a global maximizer exists it is called a *maximum likelihood estimator* (MLE); i.e. a value of $\hat{\theta}_{MLE}$ that maximizes $\mathcal{L}(\theta)$.

Let X be a finite discrete random variable with probability distribution p in a parametric model $\mathfrak{M}$ with parametrization $\phi : \Theta \to \Delta_s$. Then there exists $\theta^* \in \Theta$ such that $\phi(\theta^*) = p$. Denote by $\phi_x(\theta)$ the x coordinate of the map ϕ. Suppose that we obtain a random sample of n independent draws from X , i.e $X^{(1)} = x^{(1)}$, $X^{(2)} = x^{(2)}$,..,$X^{(n)} = x^{(n)}$. Now what if we want to infer p (and therefore θ^*) based on the sample? An often used method by frequentists to estimate parameters in a parametric model is the *maximum likelihood method*. This method consists in finding the parameter that maximizes the likelihood function, if such a parameter exists. In our case that is, maximize the probability of observing the sample $x^{(1)}, x^{(2)}, ..., x^{(n)}$ in the likelihood function with respect to the parameter θ. Since each element in the sample is independent, we can express the probability of observing the sample by

$$\mathcal{L}^*(\theta|w) = \prod_{i=1}^{n} \phi_{x^{(i)}}(\theta) = \prod_{x=1}^{s+1} \phi_x(\theta)^{w(x)}, \quad \text{where } w(x) = \#\{i : x^{(i)} = x\}. \tag{5.1}$$

Thus, to estimate θ^* we find the parameter values that maximize the likelihood function (5.1).

Note that for any $\theta^* \in \Theta$, we have $\psi(\theta^*) = p^* \in \mathfrak{M}$. Thus for the likelihood function we can find the maximum likelihood estimate $\hat{\theta}_{MLE}$ by finding a maximizer $\hat{p}_{MLE}$ constrained to the image of Θ in the probability simplex (i.e. $\hat{p}_{MLE} \in \mathfrak{M}$) and then by taking $\hat{\theta}_{MLE}$ in the fiber of $\hat{p}_{MLE}$. Thus we can think of the likelihood function as

$$L_m^*(p|w) = \prod_{x=1}^{s+1} p_x^{w(x)}, \quad p \in \mathfrak{M}.$$

This function is known as the *constrained likelihood function*. We can extend the domain of the constrained likelihood function by letting $p \in \Delta_s$. This new function is known as the *unconstrained likelihood function*.

For convenience, we take the logarithm of the likelihood function, the unconstrained likelihood function or the constrained likelihood function. The maximizer of these new functions called the *log-Likelihood*, the *unconstrained log-Likelihood*, and the *constrained log-Likelihood*, will be the same as the maximizer of their respective original functions. For example, for the likelihood function we obtain that the *log-Likelihood* is

$$\ell^*(\theta|w) = \sum_{x=1}^{s+1} w(x) \log(\phi_x(\theta)) \quad \text{for } \theta \in \Theta.$$

We observe that any sample of n independent draws from X has an empirical distribution $\hat{p}(x) = w(x)/n$. Consider the function

$$L_m(p|\hat{p}) = \prod_{x=1}^{s+1} p_x(\theta)^{\hat{p}(x)} = \prod_{x=1}^{s+1} p_x(\theta)^{w(x)/n} = \left(\prod_{x=1}^{s+1} p_x(\theta)^{w(x)} \right)^{\frac{1}{n}}.$$

Since this new function is just the n-th root of the unconstrained likelihood function, it will have the same maximizer as the unconstrained likelihood function. Thus for any $\hat{p} \in \Delta_s$ with rational entries, finding the maximizer of $L(p|\hat{p})$ is equivalent to find the maximum likelihood estimator. We can loosen the condition that $\hat{p}$ has rational entries and consider

any $\hat{p} \in \Delta_s$. We also call $L_m(p|\hat{p})$ the unconstrained likelihood function, and its logarithm the unconstrained log-Likelihood. Note that this extends naturally to the likelihood function and the constrained likelihood function.

5.2 The EM algorithm

The "EM algorithm" stands for expectation-maximization algorithm [DLR77]. It is an iterative method typically used to try to find a maximum likelihood estimator, if one exists, of parameters in statistical models where the model depends on hidden variables. The EM iteration consists of two steps; the first, the E-step, consists of computing the expectation for our hidden variable using the current estimate for the parameters. The second, the M-step, consists of the maximization of the function for the fully observed model.

For us, the EM algorithm is relevant in the sense that we want to understand how it works on Δ_{17} and $\mathcal{M}_3$; as future work we would like to determine if the EM algorithm always produces the Maximum Likelihoood Estimator, i.e. we want to determine if given a point $\hat{p} \in \Delta_{17}$, the output of the EM algorithm (EM($\hat{p}$)) will be the "best" estimate parameter θ of $\hat{p}$ in the model $\mathcal{M}_3$. This extends and complements some preliminary work from E. Allman, S. Hosten, J. Rhodes and P. Zwiernik on the maximum likelihood estimation for $2 \times 2 \times 2$ distributions. Let us describe the idea of the algorithm for our model $\mathcal{M}$.

5.3 E-step

Let $\hat{p} = \hat{p}(x_1, x_2, x_3)$ be the observed data (after normalizing, $\hat{p}$ is a distribution of format $k_1 \times k_2 \times k_3$). First, make an initial guess of the initial parameter vector θ_0; in our case $\theta_0 = \{\pi, M_1, M_2, M_3\}$, so we choose a random distribution π of size k and random Markov matrices M_i of size $k \times k_i$ for $i = 1, 2, 3$ respectively. Now we obtain the distribution $\psi(\theta_0) = p(x_1, x_2, x_3) \in \mathcal{M}$, and then we compute the expectation of the "fully observed model" $u(h, x_1, x_2, x_3) = \text{Prob}(H = h, X_1 = x_1, X_2 = x_2, X_3 = x_3)$ for the observed data $\hat{p}(x_1, x_2, x_3)$ by

$$u(h, x_1, x_2, x_3) := \hat{p}(x_1, x_2, x_3)\text{Prob}(h|x_1, x_2, x_3) = \hat{p}(x_1, x_2, x_3)\frac{\pi(h)M_1(h, x_1)M_2(h, x_2)M_3(h, x_3)}{p(x_1, x_2, x_3)}. \tag{5.2}$$

for all $(h, x_1, x_2, x_3) \in |H| \times |X_1| \times |X_2| \times |X_3|$.
To obtain $u(h, x_1, x_2, x_3)$, the assumption of conditional independence of the model is crucial to us. With it, we are able to run EM since we can express the joint distribution of the fully observed model as the product of the parameters as seen in (2.2).

5.4 M-step

For our M-step, we find θ^*, the parameter of the new estimate of $\hat{p}$ in the model. To determine π, we marginalize $u(h, x_1, x_2, x_3)$ over X_1, X_2, X_3 to give the marginal distribution of H. We note that by marginalizing over these variables what we obtain is just the distribution of the hidden variable (that is π). To determine M_1, we marginalize $u(h, x_1, x_2, x_3)$ over X_2, X_3 to find the joint distribution of H and X_1, u_{h,x_1}, and thus $M_1 = (\text{diag}(\pi))^{-1} \cdot u_{h,x_1}$. We observe that by marginalizing over X_2, X_3 we obtain the joint distribution of H and X_1, but multiplying by $\text{diag}(\pi)^{-1}$ we obtain the conditional probability of $X_1 = x_1$ given $H = h$ (that is M_1). Analogously for M_2 and M_3.

With each iteration of EM the log-likelihood function increases if possible. Then for a fixed threshold $\epsilon > 0$, if $|\theta^* - \theta_0| > \epsilon$ we set $\theta_0 = \theta^*$ and go back to the E-step; otherwise we stop. The output is the parameter θ^* and with this we can compute the corresponding probability distribution (tensor) with $\psi(\theta^*)$.

5.5 EM estimates

We were able to code the EM algorithm for our models $\mathcal{M}_2$ and $\mathcal{M}_3$. The program `EM.m` (see the appendix) has an $\epsilon = 10^{-16}$ and chooses and arbitrary probability distribution $\hat{p} \in \Delta_{q-1}$ for $q = 8, 18$ respectively.

Before we talk about the results we obtain with the EM algorithm, we will see why for any $\hat{p} \in \Delta_{q-1}$, $q = 8, 18$, we think that $\text{EM}(\hat{p})$ gives the "best" estimate of $\hat{p}$ in the model $\mathcal{M}_d$, $d = 2, 3$.

Theorem 31. *Suppose $\hat{p} \in \Delta_{q-1}$, $q = 8, 18$, and $\hat{p}$ has strictly positive entries. Then $\hat{p}_{MLE} = \hat{p}$ is a unique maximizer of the unconstrained log-Likelihood function $\ell(p|\hat{p})$. Moreover, if $\hat{p} \in \mathcal{M}_d$, $d = 2, 3$, and EM uses $\hat{p}$ as its starting point, then $EM(\hat{p}) = \hat{p}$.*

Without giving a formal proof we can observe that the proof of the first statement of the theorem is clear since we want to maximize the probability of observing the data. For the second statement of the theorem we note that since EM increases the unconstrained log-likelihood function in each iteration, if possible, EM will find a local maxima, and therefore, a local maximizer. By the first statement of the theorem this maximizer is $\hat{p}$. Therefore $\text{EM}(\hat{p}) = \hat{p}$.

But what about the MLE and $\text{EM}(\hat{p})$ for $\hat{p} \notin \mathcal{M}_d$, $d = 2, 3$? Before we talk about it, we have to introduce some concepts and statements.

Definition 32. A set $S \subset \mathbb{R}^d$ is *convex* if for any $p, p^* \in S$ and any $t \in [0, 1]$, $tp+(1-t)p^* \in S$.

Theorem 33. *The probability simplex Δ_{q-1} is convex for any $q \in \mathbb{Z}^+$.*

Proof. Let $p, p^* \in \Delta_{q-1}$ and let $t \in (0, 1)$. Note that

$$r = (r_1, r_2, .., r_q) = tp + (1-t)p^* = (tp_1 + (1-t)p_1^*, ..., tp_q + (1-t)p_q^*).$$

Thus

$$\sum_{i=1}^{q} r_i = \sum_{i=1}^{q} tp_i + (1-t)p_i^* = t + 1 - t = 1.$$

Note also that since $0 \leq p_i, p_i^* \leq 1$, then $0 \leq p_i t + (1-t)p_i^* \leq 1$. Thus $r \in \Delta_{q-1}$. Therefore Δ_{q-1} is convex. □

Definition 34. We say that a function $f : S \subset \mathbb{R} \to \mathbb{R}$ is *concave* if S is convex and for any $p, p^* \in S$,

$$tf(p) + (1-t)f(p^*) \leq f((tp + (1-t)p^*)).$$

It can be shown that the unconstrained log-likelihood function restricted to the interior of Δ_d is concave. The proof of this statement is just a matter of showing that the sum of concave functions is concave (see for example [S11]). It follows then that since $\hat{p}_i \log(p_i)$ is concave, the unconstrained log-likelihood function restricted to the interior of Δ_d is concave.

It can also be shown that if $f : S \subset \mathbb{R}^d \to \mathbb{R}$ is a concave function and S is open, then any local maximizer $s \in S$ is a global maximizer.

With these definitions and statements we are now ready to state the theorem that will describe the MLE for $\hat{p} \notin \mathcal{M}_d$.

Theorem 35. *Suppose the data $\hat{p} \in \Delta_{q-1}$, $q = 8, 18$, for the constrained optimization problem for the maximimum likelihood estimator lies outside the model $\mathcal{M}_d$, $d = 2, 3$, that is, $\hat{p} \in \Delta_{q-1} \setminus \mathcal{M}_d$, and that $\hat{p}$ has only positive entries. Then an MLE lies on the boundary of $\mathcal{M}_d$.*

Proof. Suppose that an MLE exists at some point s in the interior of $\mathcal{M}_d$. That is s is a local maximizer of the constrained likelihood $L_m(p|\hat{p})$ function. We observe that since $\dim(\mathcal{M}_d) = \dim(\Delta_{q-1})$, then any open neighbourhood of s in $\mathcal{M}_d$ contains an open neighbourhood of s in Δ_{q-1}. Thus s is also a local maximizer for the unconstrained log-Likelihood function. By Theorem 31, $\hat{p}$ is also a local maximizer of the unconstrained log-Likelihood function. Since the log-Likelihood function is concave, the segment joining s and $\hat{p}$ consists of global maximizers of the log-Likelihood function. Thus there exists some $v \in \partial\mathcal{M}_d$ that is a maximizer.

□

Remark 3. Theorem 35 suggests that for any $p \in \Delta_{q-1} \setminus \mathcal{M}_d$, $d = 2, 3$, EM should find a point on the boundary $\partial\mathcal{M}_d$. Since we do not have a complete understanding of EM's behavior about which boundary point EM is attracted to, this is a interesting question.

In the previous chapter we discussed how each irreducible component of the algebraic boundary of $\mathcal{M}_d$ ($d = 2, 3$) corresponds to the matrix parameters with zero entries. Thus given an arbitrary $\hat{p} \in \Delta_{q-1}$ ($q = 8, 18$) we compute $\theta^* = \text{EM}(\hat{p})$. By checking the zeros of M_1, M_2, M_3, and π of θ^* we can determine if the corresponding distribution of θ^* lives in the interior of the model or in some component of the algebraic boundary.

For the model $\mathcal{M}_2$ with a sample of 10000 probability distributions (tensors) randomly chosen, we can see in Table 5.1 that nearly 9% of the samples are inside the model. This matches well with our previous simulation showing that the volume of $\mathcal{M}_2$ with respect to Δ_7 is nearly 8%. We also observe that for any facet of the same dimension that appears, the counts look uniformly distributed.

Table 5.1: EM estimate attracted to various boundary strata of $\mathcal{M}_2$ for 10^4 random probability distributions.

Name	Representative Parameters	Counts
F7		898
F6		3,648
	F6a11	597
	F6a12	609
	F6b11	586
	F6b12	615
	F6c11	589
	F6c11	589
	F6c12	652
F5A		3,158
	F5Aa11b11	259
	F5Aa11c11	269
	F5Ab11c11	234
	F5Aa11b12	270
	F5Aa11c12	248
	F5Ab11c12	246
	F5Aa12b11	276
	F5Aa12c11	235
	F5Ab12c11	259
	F5Aa12b12	270
	F5Aa12c12	271
	F5Ab12c12	285
F5C		1,672
	F5Ca11a22	569
	F5Cb11b22	572
	F5Cc11c22	531
Other		4,272
Total		10,000

Table 5.2: EM estimate attracted to various boundary strata of $\mathcal{M}_3$ for 10^5 random probability distributions.

Name	Representative Parameters	Counts
F17		17
F16		351
	F16a11	49
	F16a12	53
	F16a13	47
	F16b11	40
	F16b12	45
	F16b13	49
	F16c11	37
	F16c12	31
Other		99,632
Total		100,000

We also ran the EM algorithm several times for the same $\hat{p}$ but varying the starting parameter θ_0. For example, after running EM 10000 times for the probability distribution

$$P = \left[\begin{array}{cc|cc} 0.2124 & 0.0846 & 0.4666 & 0.0986 \\ 0.0309 & 0.0383 & 0.0194 & 0.0494 \end{array}\right],$$

each with a different starting parameter, we obtain that F6a12= 9998 and F5Aa12b12= 2. This lead us to think that for this case the parameter always converges to case where M_1 has a zero on the second column.

For our model $\mathcal{M}_3$ with a sample of 100,000 probability distributions randomly chosen, we can see in Table 5.2 that the counts inside the model are fewer.

For this case we also ran the EM algorithm several times for the same $\hat{p}$ but varying the starting parameter θ_0. For example, after running EM 10,000 times for the probability distribution

$$P = \left[\begin{array}{ccc|ccc} 0.0919 & 0.0825 & 0.0447 & 0.0753 & 0.0360 & 0.0493 \\ 0.2180 & 0.0284 & 0.0313 & 0.0537 & 0.0311 & 0.0867 \\ 0.0008 & 0.0851 & 0.0070 & 0.0138 & 0.0197 & 0.0447 \end{array}\right],$$

each with a different starting parameter, we obtain that F16c11= 1529 and Other= 8471. This case is less conclusive as to whether or not the parameters are converging to a parameter where M_3 has a zero on the first column, since we do not have the detail of the facets to determine if the points in "Other" share this same property, but I believe so. Also we are surprised about how out of all the points picked outside the model, just 351 are attracted to the known facets of dimension 16 (which are the facets of greatest dimension in the model).

Chapter 6

Conclusions

It was quite surprising that the estimate of the volume of $\mathcal{M}_3$ within Δ_{17} was so small compared with the estimate of the volume of $\mathcal{M}_2$ within Δ_7. It was also very surprising that for the model $\mathcal{M}_3$, the estimates of the EM algorithm for the 10^5 random points was not concentrated in the known facets of highest dimension.

There is still plenty of work to do, like understand better the geometry of $\mathcal{M}_3$ and understand the behavior of the estimates of EM. As mentioned before we would like to determine if the estimate of the EM algorithm agrees with the maximum likelihood estimator.

Appendix

Listing 6.1: g3podtree.m

```
 function [P] = g3podtree(pie,M1,M2,M3)
%
%Ago 31 2015
q=length(pie);
t=size(M1,2);
r=size(M2,2);
s=size(M3,2);
P=zeros(t,r,s);
for i=1:t
    for j=1:r
        for k=1:s
            for p=1:q
                P(i,j,k)=pie(p)*M1(p,i)*M2(p,j)*M3(p,k)+P(i,j,k);
            end
        end
    end
end
```

Listing 6.2: params2x2x2.m

```
function [flag,M1,M2,M3,pie]=params2x2x2(P)
%Sept 7 2015
%This function receives a Tensor P of dimension 2x2x2 and gives back the
%parameters M1,M2,M3,pie (up to label
%epsilon is the criteria for rcond (determines if a function is good to work)
%flag is a variable that if get equal to 1 some matrix wans good to work
%with (computationally speaking)
epsilon=10^-6;
flag=0;
    P1oo=[P(1,1,1),P(1,1,2);P(1,2,1),P(1,2,2)];
    P2oo=[P(2,1,1),P(2,1,2);P(2,2,1),P(2,2,2)];
    Po1o=[P(1,1,1),P(1,1,2);P(2,1,1),P(2,1,2)];
    Po2o=[P(1,2,1),P(1,2,2);P(2,2,1),P(2,2,2)];
    Poo1=[P(1,1,1),P(1,2,1);P(2,1,1),P(2,2,1)];
    Poo2=[P(1,1,2),P(1,2,2);P(2,1,2),P(2,2,2)];


    if rcond(P1oo)<epsilon || rcond(P2oo)<epsilon || rcond(Po1o)<epsilon || rcond(Po2o)<
        epsilon || rcond(Poo1)<epsilon || rcond(Poo2)<epsilon
        flag=1;
    end
```

```
    P3=sum(P,3);
    P2=[P(1,1,1)+P(1,2,1),P(1,1,2)+P(1,2,2);P(2,1,1)+P(2,2,1),P(2,1,2)+P(2,2,2)];
    P1=[P(1,1,1)+P(2,1,1),P(1,1,2)+P(2,1,2);P(1,2,1)+P(2,2,1),P(1,2,2)+P(2,2,2)];


    if rcond(P1)<epsilon || rcond(P2)<epsilon || rcond(P3)<epsilon
        flag=1;
    end



   %M3(:,1)=eig(inv(P3)*Poo1);
   %M3(:,2)=eig(inv(P3)*Poo2);
   [U,V,W]=eig(inv(P3)*Poo1);
   W1=W(:,1)/sum(W(:,1));
   W2=W(:,2)/sum(W(:,2));
   M2(1,1)=W1(1);
   M2(1,2)=W1(2);
   M2(2,1)=W2(1);
   M2(2,2)=W2(2);

   pie=(inv(M2).'*P3.'*[1;1]).';
M1=inv(diag(pie))*inv(M2).'*(P3).';
M3=inv(diag(pie))*inv(M2).'*P1;



  % M1(:,1)=diag(M3*inv(P1)*P1oo*inv(M3));
  % M1(:,2)=diag(M3*inv(P1)*P2oo*inv(M3));




  % tie=[];
  % p0=inv(M1).'*P3*inv(M2);
␣␣%␣tie(1)=sum(p0(:,1));
␣␣%␣tie(2)=sum(p0(:,2));
␣␣%␣pie=tie;

␣␣␣if␣round(P,4)~=round(g3podtree(pie,M1,M2,M3),4)
␣␣␣␣␣␣␣␣flag=1;
␣␣␣end

␣␣␣if␣rcond(M1)<epsilon␣||␣rcond(M2)<epsilon␣||␣rcond(M3)<epsilon␣||␣rcond(diag(pie))<
    epsilon
␣␣␣␣␣␣␣␣flag=1;
␣␣␣end
```

Listing 6.3: `params3x3x2.m`

```
function [flag,M1,M2,M3,pie]=params3x3x2(P)
%Sept 7 2015
%This function receives a Tensor P of dimension 3x3 and gives back the
%parameters M1=3x3,M2=3x3,M3=2x2,pie=3x1 (up to label)

%epsilon is the criteria for rcond (determines if a function is good to work)
%flag is a variable that if get equal to 1 some matrix wans good to work
%with (computationally speaking)
epsilon=10^-6;
flag=0;
    %P..+
    P3=sum(P,3);
    P1=[P(1,1,1)+P(2,1,1)+P(3,1,1),P(1,1,2)+P(2,1,2)+P(3,1,2);
        P(1,2,1)+P(2,2,1)+P(3,2,1),P(1,2,2)+P(2,2,2)+P(3,2,2);
        P(1,3,1)+P(2,3,1)+P(3,3,1),P(1,3,2)+P(2,3,2)+P(3,3,2)];
    if rcond(P3)<epsilon
        flag=1;
    end
    %Pi..
    P1oo=[P(1,1,1),P(1,1,2);P(1,2,1),P(1,2,2)];
    P2oo=[P(2,1,1),P(2,1,2);P(2,2,1),P(2,2,2)];
    P3oo=[P(3,1,1),P(3,1,2);P(3,2,1),P(3,2,2)];

    if rcond(P1oo)<epsilon || rcond(P2oo)<epsilon || rcond(P3oo)<epsilon
        flag=1;
    end

    %P.i.
    Po1o=[P(1,1,1),P(1,1,2);P(2,1,1),P(2,1,2)];
    Po2o=[P(1,2,1),P(1,2,2);P(2,2,1),P(2,2,2)];
    Po3o=[P(1,3,1),P(1,3,2);P(2,3,1),P(2,3,2)];


    if rcond(Po1o)<epsilon || rcond(Po2o)<epsilon || rcond(Po3o)<epsilon
        flag=1;
    end

    %P..i
    Poo1=[P(1,1,1),P(1,2,1),P(1,3,1);P(2,1,1),P(2,2,1),P(2,3,1);P(3,1,1),P(3,2,1),P(3,3,1)];
    Poo2=[P(1,1,2),P(1,2,2),P(1,3,2);P(2,1,2),P(2,2,2),P(2,3,2);P(3,1,2),P(3,2,2),P(3,3,2)];

      if rcond(Poo1)<epsilon || rcond(Poo2)<epsilon
        flag=1;
      end

   %M3(:,1)=eig(inv(P3)*Poo1);
   %M3(:,2)=eig(inv(P3)*Poo2);
```

```
   [U,V,W]=eig(inv(P3)*Poo1);
   W1=W(:,1)/sum(W(:,1));
   W2=W(:,2)/sum(W(:,2));
   W3=W(:,3)/sum(W(:,3));
   M2(1,1)=W1(1);
   M2(1,2)=W1(2);
   M2(1,3)=W1(3);
   M2(2,1)=W2(1);
   M2(2,2)=W2(2);
   M2(2,3)=W2(3);
   M2(3,1)=W3(1);
   M2(3,2)=W3(2);
   M2(3,3)=W3(3);

   pie=(inv(M2).'*P3.'*[1;1;1]).';
M1=inv(diag(pie))*inv(M2).'*(P3).';
M3=inv(diag(pie))*inv(M2).'*P1;

    if rcond(M1)<epsilon || rcond(M2)<epsilon || rcond(M3(1:2,1:2))<epsilon || rcond(M3(2:3
        ,1:2))<epsilon || rcond(diag(pie))<epsilon
        flag=1;
    end
```

Listing 6.4: `Sto2x2x2.m`

```
function [SZ,RE,IM]  =Sto2x2x2(n)
%Sept 7 2015
%Gives the percentage of the simplex that acutally comes from the model of the Markov chain
%Has just one variable, 'n' the number of sample

%a is the oarameter of the dirichlet distribution, in this case a vector of ones gives a
    uniform dist, the size of a is the dimension of the simplex
a=[1,1,1,1,1,1,1,1];

% This next step generates the random point in the simplex with the dirichlet dist
p = length(a);
r = gamrnd(repmat(a,n,1),1,n,p);
r = r ./ repmat(sum(r,2),1,p);

s=0;
re=0;
im=0;
%P carries the point of the simplex as a tensor
tic
for i=1:n
    P(:,:,1)=[r(i,1),r(i,2);r(i,3),r(i,4)];
    P(:,:,2)=[r(i,5),r(i,6);r(i,7),r(i,8)];

   %flag commes from the rcond condition if something used was not good
```

```
   [flag,M1,M2,M3,pie]=params2x2x2(P);

   if flag==1
       continue
   end

    if  sum(sum(imag(M1)==0))==4 &&  sum(sum(imag(M2)==0))==4 && sum(sum(imag(M3)==0))==4 &&
         sum(sum(imag(pie)==0))==2

    if sum(sum(M1<1))==4 && sum(sum(M2<1))==4 && sum(sum(M3<1))==4 && sum(sum(pie<1))==2
            if sum(sum(M1>=0))==4 && sum(sum(M2>=0))==4 && sum(sum(M3>=0))==4 && sum(sum(pie
                >0))==2
                 s=s+1;

            else
                re=re+1;

            end
         else
          re=re+1;
         end
    else
      im=im+1;
    end

end


toc
   SZ=100*s/n;
   RE=100*re/n;
   IM=100*im/n;
```

Listing 6.5: `Sto3x3x2.m`

```
function [SZ,RE,IM] =Sto3x3x2(n)
%Sept 7 2015
%Gives the percentage of the simplex that acutally comes from the model of the Markov chain
%Has just one variable, 'n' the number of sample

%a is the oarameter of the dirichlet distribution, in this case a vector of ones gives a
    uniform dist, the size of a is the dimension of the simplex
a=[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1];

% This next step generates the random point in the simplex with the dirichlet dist
p = length(a);
r = gamrnd(repmat(a,n,1),1,n,p);
r = r ./ repmat(sum(r,2),1,p);
```

```
s=0;
re=0;
im=0;
%P carries the point of the simplex as a tensor
tic
for i=1:n
    P(:,:,1)=[r(i,1),r(i,2),r(i,3);r(i,4),r(i,5),r(i,6);r(i,7),r(i,8),r(i,9)];
    P(:,:,2)=[r(i,10),r(i,11),r(i,12);r(i,13),r(i,14),r(i,15);r(i,16),r(i,17),r(i,18)];

    %flag commes from the rcond condition if something used was not good
   [flag,M1,M2,M3,pie]=params3x3x2(P);

   if flag==1
       continue
   end

    if  sum(sum(imag(M1)==0))==9 &&  sum(sum(imag(M2)==0))==9 && sum(sum(imag(M3)==0))==6 &&
         sum(sum(imag(pie)==0))==3

          if sum(sum(M1<1))==9 && sum(sum(M2<1))==9 && sum(sum(M3<1))==6 && sum(sum(pie<1))
              ==3
            if sum(sum(M1>0))==9 && sum(sum(M2>0))==9 && sum(sum(M3>0))==6 && sum(sum(pie>0)
                )==3

                s=s+1;

            else
                re=re+1;

            end
         else
          re=re+1;
         end
    else
      im=im+1;
    end

end


toc
   SZ=100*s/n;
   RE=100*re/n;
   IM=100*im/n;
```

Listing 6.6: `EM.m`

```
function [Phat,P,ipie,iM1,iM2,iM3,pie,M1,M2,M3]=EM(rank)
%Oct 16 2015

%epsilon is our stopping criteria between the difference of the estimated
%parameters
epsilon=10^-16;
h=rank;
% m1,m2,m3 are the parameters of the previous estimation
m1=zeros(h);
m2=zeros(h);
m3=zeros(h);
%piesin=zeros(1,3);
%This next step choose a random point from the space
r=Dirichlett(rank);
% this step organizes it as a tensor
if rank==3
    for i=1:1
        Phat(:,:,1)=[r(i,1),r(i,2),r(i,3);r(i,4),r(i,5),r(i,6);r(i,7),r(i,8),r(i,9)];
        Phat(:,:,2)=[r(i,10),r(i,11),r(i,12);r(i,13),r(i,14),r(i,15);r(i,16),r(i,17),r(i,18)
            ];
    end
else if rank==2
     for i=1:1
        Phat(:,:,1)=[r(i,1),r(i,2);r(i,3),r(i,4)];
        Phat(:,:,2)=[r(i,5),r(i,6);r(i,7),r(i,8)];
     end
    end
end

[x1,x2,x3]=size(Phat);
%n=x1+x2+x3;
%D=P/n;
%This are the first choice of parameters to aproximate, they are chose
%randomly
pie=rand(1,h);
M1=rand(h,x1);
M2=rand(h,x2);
M3=rand(h,x3);

pie=pie/sum(pie);
M1=inv(diag(sum(M1,2)))*M1;
M2=inv(diag(sum(M2,2)))*M2;
M3=inv(diag(sum(M3,2)))*M3;

ipie=pie;
iM1=M1;
iM2=M2;
iM3=M3;
```

```
Z=zeros(h,x1,x2,x3);
u=zeros(h,x1,x2,x3);
P=zeros(x1,x2,x3);

%this P is the tensor resulted of our chosen parameters
 P=g3podtree(pie,M1,M2,M3);

%%%%%%%%
 %E-step :
 %%%%%%%%
for loop=1:50000
     % the nex loop commented is another way to get P
    % for i=1:2
%    Petit(:,:,i)=M1'*diag(pie'.*M3(:,i))*M2
% end
% vector with only i-th entry of pie
    for i=1:h
        v=pie.*(i==[1:h]);
        %This array of posterior probabilities of X_0 for all possible
        %observations that is , for fixed (ii,jj,kk) u(:,ii,jj,kk)=Prob(X_0|ii,jj,kk)
        for j=1:x3
            Z(i,:,:,j)=(M1'*diag(v'.*M3(:,j))*M2)./P(:,:,j);
        end
     %this step get u(x,h)=u(x)p(h|x;parameters). This is just the observed data times time
         the conditional probability Prob(X_0|ii,jj,kk)
    u(i,:,:,:)=squeeze(Z(i,:,:,:)).*Phat;
    end



%%%%%%%%
 %M-step :
 %%%%%%%%
 %estimating new mixing parameters
    pie=sum(sum(sum(u,4),3),2)';
␣␣␣␣Div=inv(diag(pie));␣%␣for␣dividing␣by␣row␣sums
␣␣␣␣A=sum(sum(u,4),3);␣%marginalize␣to␣X_0,␣X_1␣variables
␣␣␣␣M1=Div*A;␣␣%␣divides␣each␣row␣by␣row␣to␣get␣M1

␣␣␣␣if␣rank==3
␣␣␣␣Bt=sum(sum(u,4),2);␣%marginalize␣to␣X_0,␣X_2␣variables
␣␣␣␣B(:,1)=Bt(:,:,1);
␣␣␣␣B(:,2)=Bt(:,:,2);
␣␣␣␣B(:,3)=Bt(:,:,3);
␣␣␣␣M2=Div*B;
␣␣␣␣Ct=sum(sum(u,3),2);␣%marginalize␣to␣X_0,␣X_3␣variables
␣␣␣␣C(:,1)=Ct(:,:,1,1);
␣␣␣␣C(:,2)=Ct(:,:,1,2);
␣␣␣␣M3=Div*C;
```

```
    else if rank==2
        Bt=sum(sum(u,4),2); %marginalize to X_0, X_2 variables
    B(:,1)=Bt(:,:,1);
    B(:,2)=Bt(:,:,2);
    M2=Div*B;
    Ct=sum(sum(u,3),2); %marginalize to X_0, X_3 variables
    C(:,1)=Ct(:,:,1,1);
    C(:,2)=Ct(:,:,1,2);
    M3=Div*C;
        end
    end




    P=g3podtree(pie,M1,M2,M3);

    %Stoping criteria check that the difference entry by entry between the new parameters
    %and the old parameters is less than epsilon
    check=0;
    for i=1:h
        for j=1:h
            if abs(M1(i,j)-m1(i,j))<epsilon
                check=check+1;

            end
        end
    end

    for i=1:h
        for j=1:h
            if abs(M2(i,j)-m2(i,j))<epsilon
                check=check+1;

            end
        end
    end

    for i=1:h
        for j=1:2
            if abs(M3(i,j)-m3(i,j))<epsilon
                check=check+1;

            end
        end
    end
    % The new parameter changed to old
    m1=M1;
```

```
    m2=M2;
    m3=M3;


    if rank==3
        if check==24
            loop;
            return
        end
    elseif rank==2
        if check==12
            loop;
            return
        end
    end
end
```

Listing 6.7: `bndstrata.m`

```
function bndstrata(filename)
% 11/8/15
% check pdf file Boundary stata. This function receives a file generated in
% the function generatebndstata.
%filename='Bnstratafixed2.txt';
A = dlmread(filename);
n=size(A,1);
F7=0;
F6=zeros(1,6);
F5a=zeros(1,12);
F5b=zeros(1,12);
F5c=zeros(1,3);
F4a=zeros(1,8);
F4b=zeros(1,24);
F3p=0;
F3a=zeros(1,6);
F3b=zeros(1,24);
other=0;
% the archive filename is a matrix that each row corresponds to the row of
% the pdf file Boundary strata for 2x2x2

tic
for i=1:n
   k=0;
   for j=0:2
       % Z is a matrix that tells where are the zeros or ones in the
       % parameters M1 M2 M3
       % Z(j+1,:) is a vector correspondent to the parameter Mi, it tells
       % whethere there is a zero or a 1 in Mi and if is either in the
       % first row or the second row of Mi
```

```
      Z(j+1,:)=[ A(i,17+2*j)==0 , A(i,18+2*j)==0 , A(i,17+2*j)==1 ,  A(i,18+2*j)==1];
      %z tells the number of zeros  per parameters M1 M2 M3
      z(j+1)= sum(Z(j+1,:));
      % k is the total number of zeros in M1 M2 M3
      k=k+sum(z(j+1));
  end

  % check cases acording to number of zeros
 switch k

     %first we check if there are no zeros in the parameters
  case 0
     %We check first if pie has a zero or not
      if A(i,16)==1 || A(i,16)==0
          F3p=F3p+1;
       % if pie does not have a zero then that tensor lives in F7
      else
          F7=F7+1;
      end
      %Case there is exactly one zero
  case 1
      %In the next cycle we find in which parameter there zero lives
      for j=0:2
          % in the next step we check each parameter to find a zero of a
          % one in the first column of the parameter,
          if A(i,2*(j)+17)==0 || A(i,2*(j)+17)==1 || A(i,2*(j)+18)==0 || A(i,2*(j)+18)==1
              % we save the position of the zero acroding to de Boundary
              % strata file
            pos=2*j+1+(A(i,2*j+17)==1)+ (A(i,2*(j)+18)==1);
            F6(1,pos)=F6(1,pos)+1;
          end
      end

     % case where there are 2 zeros
   case 2
       %arriba and abajo saves tells whether the zeros are in the first
       %row (arriba) or in the second (abajo) and is a vector with 1's
␣␣␣␣␣␣␣␣%and␣2's it will be clarified later
       arriba=zeros(1,3);
       abajo=zeros(1,3);

      % in this cycle we try to find the zeros of each parameter
          for j=1:3
               %In this step we find if there are 2 zeros in one parameter
               if z(j)==2
                   % here we check if the zero are in the same column or
                   % not. If Z(j,1)==Z(j,2) that means that the first row
                   % has either two zeros or two ones
                   if Z(j,1)==Z(j,2)
```

```
                if Z(j,1)==1
                    F3a(1,2*j-1)=F3a(1,2*j-1)+1;
                else
                    F3a(1,2*j)=F3a(1,2*j)+1;
                end
                % if there the column does not have two zeros or
                % ones in the first column then  we end in this
                % case
            else
                F5c(j)=F5c(j)+1;
            end



          % case where there are two zeros, each in a different
          % parameter
        elseif z(j)==1
                 %we check if the zeros where the zeros are in the first column (
                     arriba) or
                 %in the second column (abajo). arriba and abajo
                 %are vectors of size three with 0,1,2
                 %acording to if there is a zero in the first
                 %row. arriba(i)=0 if there is no zero in the
                 %first row of Mi, arriba(i)=1 if Mi has one
                 %zero in the first row and first column and
                 %arriba(i)=2 if M2 has one zero in the first row
                 %and second column.

                 % in this step we verify if the zero is arriba or
                 % abajo  recall that Z(j,1) has a 1 if the
                 % M(1,1)=0 and Z(j,3)=1 if M(1,1)=1
                 if Z(j,1)==1 || Z(j,3)==1
                     if  Z(j,1)==1
                         arriba(j)=1;
                     else
                         arriba(j)=2;
                     end
                     %analogously
                  else if Z(j,2)==1
                         abajo(j)=1;
                     else
                         abajo(j)=2;
                     end
                  end


        end
    end
```

```
        % if arriba or abajo are zeros that means we end in
        % case F5A
       if sum(arriba)==0 || sum(abajo)==0

           if sum(arriba)==0
               %buenvector is just the vector that is not zero
               buenvector=abajo;
           else
               buenvector=arriba;
           end
           % finds the correspondent place in the list "long
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣%␣name"

           sv=sum(buenvector);
            %since buenvector is a vector with 0,1,2 then its
            %and one entry is zero (becuase we only have one
            %zero), then sv is equal to either 2,3 or 4

           if sv==2

               pos=buenvector(1)+buenvector(2)*2+buenvector(3)*3-2;
               F5a(pos)=F5a(pos)+1;
           elseif sv==3
               if buenvector(1)==1 || buenvector(1)==0 && buenvector(2)==1
                  pos=buenvector(1)+buenvector(2)*2+buenvector(3)*3-1-(
                      buenvector(2)==0)-(buenvector(1)==0);
                  F5a(pos)=F5a(pos)+1;

               else
                  pos=buenvector(1)+buenvector(2)*2+buenvector(3)*3+3-(
                      buenvector(1)==0);
                  F5a(pos)=F5a(pos)+1;
               end

           elseif sv==4
               pos=buenvector(1)+(buenvector(2))*2+(buenvector(3))*3+4-(
                   buenvector(2)==0)-2*(buenvector(1)==0);
               F5a(pos)=F5a(pos)+1;
           end


       else
           %Case when arriba and abajo are not zero. The same
           %type of analyisis.
          buenvector=arriba+abajo;
           sv=sum(buenvector);
           if sv==2
               pos=buenvector(1)+buenvector(2)*2+buenvector(3)*3-2;
               F5b(pos)=F5b(pos)+1;
```

```
                    elseif sv==3
                        if buenvector(1)==1 || buenvector(1)==0 && buenvector(2)==1
                           pos=buenvector(1)+buenvector(2)*2+buenvector(3)*3-1-(
                               buenvector(2)==0)-(buenvector(1)==0);
                           F5b(pos)=F5b(pos)+1;

                        else
                           pos=buenvector(1)+buenvector(2)*2+buenvector(3)*3+3-(
                               buenvector(1)==0);
                           F5b(pos)=F5b(pos)+1;
                        end

                    elseif sv==4
                        pos=buenvector(1)+(buenvector(2))*2+(buenvector(3))*3+4-(
                            buenvector(2)==0)-2*(buenvector(1)==0);
                        F5b(pos)=F5b(pos)+1;
                    end
                end




   %case when we have 3 zero s
case 3
    % the first case is when the three parameters each has a zero
    if z(1)==1 && z(2)==1 && z(3)==1

            % in this part we check the position according to the file
            % boundary strata
        if (Z(1,1)==1 || Z(1,3)==1) &&  (Z(2,1)==1 || Z(2,3)==1) && (Z(3,1)==1 || Z
            (3,3)==1)
             pos=4*A(i,17)+1+2*A(i,19)+A(i,21);
             F4a(pos)=F4a(pos)+1;
        elseif (Z(1,2)==1 || Z(1,4)==1) &&  (Z(2,2)==1 || Z(2,4)==1) && (Z(3,2)==1 || Z
            (3,4)==1)
             pos=4*A(i,18)+1+2*A(i,20)+A(i,22);
             F4a(pos)=F4a(pos)+1;
        else
            other=other+1;
        end
    end
    % case one parameter has two zeros and other parameter has one zero
   for j=1:3
       if z(j)==2
           % case when the parameter that has two zeros has them in the
           % same colum
         if Z(j,1)==Z(j,2) || Z(j,3)==Z(j,4)
             pos=(j-1)*8+1+4*(Z(j,3)==1)+2*z(mod(j+1,3)+1)+Z(mod(j+1,3)+1,4)+Z(mod(j
                 +1,3)+1,3)+Z(mod(j,3)+1,4)+Z(mod(j,3)+1,3);
```

```
                F3b(pos)=F3b(pos)+1;


            else
                % case when the parameter that has two zeros each in
                % different column
                pos=(j-1)*8+1+4*z(mod(j+1,3)+1)+2*(Z(mod(j+1,3)+1,4)+Z(mod(j+1,3)+1,3))+Z(
                    mod(j,3)+1,4)+Z(mod(j,3)+1,3)+Z(mod(j+1,3)+1,4)+Z(mod(j+1,3)+1,3);
                F4b(pos)=F4b(pos)+1;
            end
        end
    end



     %other cases
  otherwise
    other=other+1;
 end



end
toc
 F7;
 F6;
 F5a;
 F5b;
 F5c;
 F4a;
 F4b;
 F3p;
 F3a;
 F3b;
 other;

 %printing is a function that orders the resulting vectors and asigns its
 %value printing according to its position
 printing(F7, F6,F5a,F5b,F5c,F4a,F4b,F3p,F3a,F3b,other)
```

[bndstarta3x3x2.m]

Listing 6.8: `bndstarta3x3x2.m`

```
function bndstrata3x3x2(filename)
% 11/8/15
% check pdf file Boundary stata. This function receives a file generated in
% the function generatebndstata.
%filename='Bnstratafixed2.txt';
```

```
A = dlmread(filename);
n=size(A,1);
F17=0;
F16=zeros(1,8);
other=0;


for i=1:n
   k=0;
   for j=1:18
       Z(j)=[(A(i,37+j)==0)];
   end
   for j=19:21
       Z(j)=[(A(i,37+j)==0)+(A(i,37+j)==1)];
   end

  k=sum(Z);
   % check cases acording to number of zeros
  switch k
      %first we check if there are no zeros in the parameters
   case 0
      %We check first if pie has a zero or not
       if A(i,35)==1 || A(i,35)==0
           other=other+1;
        % if pie does not have a zero then that tensor lives in F17
       else
           F17=F17+1;
       end
       %Case there is exactly one zero
   case 1
      for j=1:18
          if Z(j)==1
              pos=ceil(j/3);
              F16(pos)=F16(pos)+1;
          end
      end
      for j=19:21
          if Z(j)==1
             if A(i,56)==1 ||  A(i,57)==1 ||  A(i,58)==1

             F16(8)=F16(8)+1;
             else
                 F16(7)=F16(7)+1;
             end
          end
      end
      otherwise
          other=other+1;
```

```
  end
end

if sum(F17)~=0
    fprintf('␣F17=␣%d␣\n',F17)
end
if sum(F16)~=0
     f16={};
     valores={'F6a11=','F6a12=','F6a13=','F6b11=','F6b12=','F6b13=','F6c11=','F6c12='};
    for i=1:8
        if F16(i)~=0

            f16{i}=strcat(valores{i},int2str(F16(i)));

        end
    end
    f16(cellfun('isempty',f16)) = [];                      % remove the empty cells
    fprintf('\n␣%s\n','F16')
    fprintf('␣%s\n',f16{:})
 end


 if other~=0
     fprintf('\n␣Other=␣%d\n',other);
 end
```

References

[ARSV14] E. Allman, J. Rhodes, E. Stanghellini and M. Valtorta. Parameter Identifiability of Discrete Bayesian Networks with Hidden Variables. *Journal of Causal Inference*, 2014.

[ART14] E. Allman, J. Rhodes and A. Taylor. A semialgebraic model description of the general Markov model on phylogenetic trees. *SIAM J. Discrete Math*, Vol. 28, No. 2, pp. 736-755, 2014.

[ARSZ15] E. Allman, J. Rhodes, B. Sturmfels and P. Zwiernik. Tensors of nonnegative rank two. *Linear algebra and its applications*, Vol. 473, pp. 37-55, 2015.

[CLO07] D. Cox, J. Little and D. O'Shea. Ideals, Varities, and Algorithms. *Springer*, (3), 2007.

[DLR77] A. Dempster, N. Laird, D. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 39, No. 1. pp. 1-38, 1977.

[KBJ00] S. Kotz, N. Balakrishnan, and N. Johnson. Continuous Multivariate Distributions. Volume 1: Models and Applications. *New York: Wiley*, Chapter 49: Dirichlet and Inverted Dirichlet Distributions, 2000.

[MSZTL13] R. Mourad, C. Sinoqut, N. Zhang, L. Tengfei, P Leray. A Survey on Latent Tree Models and Applications. *Journal of machine learning research*,Vol. 47, pp.157-203, 2013.

[MTAW00] J. Myung, V. Tan, A. Anandkumar, A. Willsky. Learning Latent Tree Graphical Models. *Journal of machine learning research*,Vol. 1, pp.1-48, 2000.

[S11] B. Simon. Convexity: An Analytic Viewpoint. *Cambridge University Press*, Cambridge, 2011.

[ZS11] P. Zwiernik and J. Smith. Implicit inequality constrains in a binary tree model. *Electronic Journal of Statistics*, Vol. 5, pp. 1276-1312, 2011.

[ZS12] P. Zwiernik and J. Smith. Tree cumulants and the geometry of binary tree models. *Bernoulli*, Vol. 18, No. 1, pp. 290-321, 2012.

[Z16] P. Zwiernik. Tree models. *In preparation*, 2016.