Supervised and Unsupervised Model-Based Clustering with Variable Selection

A thesis presented for the degree of

Doctor of Philosophy of Imperial College London

by

Alberto Maria Cozzini

Department of Mathematics

Section of Statistics

Imperial College London

180 Queen's Gate

London SW7 2BZ

OCTOBER 2011

I certify that this thesis, and the research to which it refers, are the product of my own work, and that any ideas or quotations from the work of other people, published or otherwise, are fully acknowledged in accordance with the standard referencing practices of the discipline.

Signed:

Copyright

Copyright in text of this thesis rests with the Author. Copies (by any process) either in full, or of extracts, may be made **only** in accordance with instructions given by the Author and lodged in the doctorate thesis archive of the Imperial College London central library. Details may be obtained from the Librarian. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the Author.

The ownership of any intellectual property rights which may be described in this thesis is vested in Imperial College London, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement. Further information on the conditions under which disclosures and exploitation may take place is available from the Imperial College London registry.

To my family and friends who have lived this journey with me.

"Tamdiu discendum est, quamdiu nescias"
Lucius Annaeus Seneca

Abstract

The thesis tackles the problem of uncovering hidden structures in high-dimensional data in the presence of noise and non informative variables. It proposes a supervised and an unsupervised mixture models that select the relevant variables and are robust to measurement errors and outliers.

Within the class of unsupervised clustering models we extend variable selection to the family of Student's t mixture models. While t distributions are naturally robust to noise and extreme events, sparsity is achieved by imposing regularization on the location and dispersion parameters. An EM algorithm is implemented to return the maximum likelihood estimate of the model parameters given the added penalty term. To further asses the contribution of each variable we propose a resampling procedure that ranks the variables according to their selection probability.

Supervised clustering is implemented in a Bayesian framework. The model assumes a mixture of Lasso type regressions with t-distributed errors. While the Lasso representation of the normal linear model imposes regularization on the regression coefficient, variable selection is explicitly modelled by a latent binary indicator variable. The model relies on particle Markov chain Monte Carlo algorithm to approximate the posterior distribution of the parameters of interest.

To highlight the properties and advantages of the proposed models, two real life problems are considered. The first one requires us to identify subtypes of breast cancer tumors by grouping patients based only on their gene expression levels when only few of the thousands genes are informative. In the second case our aim is to cluster different financial markets spanning several macro sectors and explain their trading performance only on the basis of the observed statistical features of their price dynamics.

Acknowledgements

I would like to thank my supervisors Dr. Giovanni Montana and Dr. Ajay Jasra for their generous donation of time, effort and patience.

I would also like to thank Dr. Stefano Luzzatto who gave me the chance to enrol in the PhD programme at Imperial and thank Dimitri Vvedensky, John Gibbons and David Hand who made it possible for me to continue in the programme.

I am grateful to AHL Research and Dr. Anita Grigoriadis for providing the data I analyze in this thesis.

Would finally like to thank all my past teachers who taught me more than what I could have learnt from the books.

Table of contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

The process of learning is a natural and vital part of the growth and development of every person. On a larger scale, the pursuit of knowledge drives the progress of every human civilization. In a modern knowledge-based society the advancement of scientific disciplines relies on a rigorous learning process which can follow a theoretical deduction approach or induction from observations.

In the same way a single person learns from their own life experiences, scientific knowledge can be acquired by moving from the observation of a series of events to the underlying principles that explain them. Statistical inference plays a pivotal role in the inductive learning process. Given a dataset of observations, statistics provides the theory and tools to draw inference about the general rule behind the specific events recorded.

In the learning process that goes from the repeated observation of multiple phenomena to the formulation of a general rule, clustering provides a rigorous method to distinguish between the accidental specific features of each datapoint and the essential common elements that characterize an homogeneous class of events. Conversely, while it allows us to group together similar samples, it highlights the critical differences that separates one cluster from another.

Equally important, in order to achieve an accurate representation of the phenomena, is to recognise, of all observed features, which are really relevant for the understanding of the underlying general principles. In statistics this part of the

learning process is formalized as a variable selection procedure. It allows us to reduce the dimensionality of the dataset that we want to model and ultimately yields a more clear and parsimonious derivation of the general rule we are interested in.

Another important aspect of the learning process is to investigate the link between two separate events where we presume there is a cause-effect relation. Given a collection of paired independent and dependent datapoints, the primary task of the clustering procedure is again to identify groups of similar samples. In this case, the process of partitioning a heterogeneous population of explanatory variables and the corresponding response variables takes the name of supervised clustering. The variable selection procedure is still a critical step towards a better understanding and a more accurate representation of the causality relation.

In the present thesis we will propose two separate probabilistic models, one implementing an unsupervised clustering approach and the other implementing a supervised clustering approach. We illustrate the validity of the statistical methods we introduce by applying them to two diverse real life problems. In both cases we will highlight the importance of being able to select only the relevant variables and how this procedure improves the performance of the models and interpretation of the results.

Before proceeding any further, we should stress the point that any learning process, even the most rigorous and precise, has to be combined with a sound interpretation of the results and intelligent elaboration of the information we acquire. Whilst statistics is the epitome of the experimental learning process, we need to be aware of the limits of the knowledge acquired from past experience, as the chicken, who got to trust the farmer that saw feeding him day after day till he got slaughtered, can testify (Russell, 1912).

1.1 Real Life Problems

To illustrate how the two clustering models we propose have a general validity and are suitable to represent real data, we have chosen to investigate two real life problems from two very distant application areas.

In one case we start our analysis from the recorded gene expression levels of a cohort of patients diagnosed with breast cancer. The aim of the study is to assess whether we can identify separate subtypes of carcinoma. In particular, we are interest in identifying which genes might be associated with the insurgency of each subtype.

In the second case we intend to propose an alternative partition of the financial markets that better suits a systematic investment approach. From the observation of the price dynamics of each market we extract an array of explanatory variables. We then compute for each market a measure of the performance of a simple systematic trading strategy, which is effectively the response variable we would like to explain.

To fully appreciate the properties of the models we develop in this thesis, we shall first discuss in detail the main features of each of the datasets that motivated our work.

1.1.1 Clustering Microarray data and Gene Selection

Since the scientific breakthrough that allowed to decode human genome, over the past decade rapid developments in genomic and other molecular research technologies have combined to produce a tremendous amount of information related to molecular biology.

Microarray gene expressions studies are routinely carried out to measure the transcription levels of an organism's genes. Still, for the main part, the exact function of each gene is unknown and this challenge has led to the exponential growth of bioinformatics. One of the many bioinformatics tasks is to learn from the different gene expression levels observed in a population and improve the understanding of the biological processes.

In the present thesis we will investigate the expression levels of genes from patients diagnosed with breast cancer. The objective of the investigation is to isolate naturally occurring groups of patients with similar gene expression patterns. This learning process should lead to the discovery of molecular fingerprints that

define subtypes of the disease with clinically distinct prognosis and requiring a more targeted treatment.

We should also note that in microarray studies it is expected that not all the gene expression measurements will necessarily contribute equally to the identification of distinct sub-groups of samples. Even when real clusters exist and are well separated, it is often the case that only a subset of genes will have expression levels that significantly vary across groups. Failing to identify the truly informative genes may yield inaccurate clustering results because the non-informative genes will mask the underlying structure of the data.

In order to extract as much information as possible from the data and draw reliable inference about the biological implication of the results, we intend to propose a model that responds to the specific challenges posed by the microarray records we set out to analyze:

- Few Biological Samples
- Very High Dimensional Observations
- Non Informative Variables
- Extreme Observations/Heavy Tail Distribution
- Measurement Noise

A model that satisfies these requirements would put us in the position to pursue more focused investigation on each cancer subtype. Moreover, once we have identified the informative genes, we can explore the relation between their over/under expression and the specific pathology we observe. This insight will hopefully lead to a more targeted and effective treatment.

1.1.2 The Need for Segmenting Financial Markets

In the financial literature it is common practice to group markets into macro sectors based on the type and nature of the good exchanged. Practitioners operating in

financial markets adhere to this convention and consider each sector as a separate area of expertise. This approach is reasonable for fundamental investors who have to be knowledgeable on the underlying factors driving demand/offer and have to elaborate the relevant information as news become public.

A partition of the markets that mirrors the macro sectors is less obviously suitable to systematic traders who take investment decisions based on algorithms which depend only on the evolution of prices. Under these circumstances, developing and optimizing a quantitative strategy on a sector by sector basis seems rather arbitrary. This is because the only input considered when engineering the strategy is the time series of prices whose behaviour is not necessarily a function of the sector. A clustering method which is more consistent with a systematic and objective approach, should identify homogeneous clusters of markets that share similar price dynamics characteristics.

Our approach starts by selecting, across all sectors, those major financial markets for which we have records spanning up to twenty years of trading. Under the assumption that all the relevant information about a market can be extracted from the historical prices, we then compute for each market the summary statistics that measure the critical features of the distribution and the temporal dependence of time series of returns.

The learning process can follow an unsupervised approach and, as for the gene data analysis, propose a clustering model that differentiates between relevant and irrelevant variables. By selecting only the statistics that really help to characterize each market, we achieve a more accurate partition of the markets in homogeneous clusters.

In a supervised learning framework, the same statistics of the price dynamics can be seen as explanatory variables that can help us understand why the trading performance is different across markets. When we apply the same basic trading algorithm to every market, we observe that the risk-adjusted profit we obtain is not consistent across markets. The supervised model we propose should be able to regress the profitability of the trading algorithm on some of the features we record for each market.

Assuming we achieve a more accurate partition of the markets, we are in an ideal position to develop a systematic trading strategy that better suits the markets within each group. A strategy that has been optimized on a market by market basis would likely be overfitted and would not have enough back testing data. If, instead, we devise a trading algorithm that consistently performs on a group of markets, we are bound to obtain a more robust and convincing result. At the same time, the significant features that are responsible for driving the clustering process give us an insight on the critical aspects of price dynamics that should be exploited by the trading strategy.

In order to reach credible conclusions about how to partition markets and what are the informative features of the price dynamics, we need a clustering method which is able to address the following issues:

- Supervised Partitioning
- Small Number of Samples
- Fewer Observations than Explanatory Variables
- Non Informative Variables
- Outliers
- Measurement Noise

If we succeed in proposing a model whose performance is not hindered by these issues, we will have increased confidence in the trading strategy that we develop based on the outcome of clustering and variable selection process. By being able to implement a more targeted strategy on each group of markets, we should achieve better investment returns.

1.2 Contributions

In an effort to meet the very specific requirement of the two real life problems we just presented, we review the existing literature on supervised and unsupervised learning methods. Whilst we find that mixture models are a flexible and

effective method to cluster a heterogeneous population, like the one we suspect is represented in the two datasets, we did not find a specific model that would exactly match our requirements. In order achieve a more fitting and stringent representation we resolve to propose the following two new mixture models.

The first model we introduce to perform unsupervised clustering is a penalised mixture of Student's t distributions. Mixture models have been widely implemented to describe high dimensional datasets, such is the case with microarray data, because they can fairly easily fit a parametric density function for each component irrespective of the number of datapoints. This property is essential in our investigation given that the two datasets we model only have a small number of samples. We assume Student's t density components because they can achieve slower exponential tail decay, thus yielding heavier tails and making the model more robust to measurement noise and the extreme observations we noted in gene and financial data.

While the majority of algorithms for unsupervised data partitioning use all the variables that describes the objects to be clustered, we address the problem of detecting clusters that only exist in a reduced number of dimensions by introducing variable selection procedure to t mixture models. This procedure allows us to automatically exclude from the model those genes and market returns statistics that are not informative.

Maximum likelihood estimation approaches achieve variable selection by imposing penalty constraints on the likelihood which has the effect of shrinking some parameters to common values. We propose a joint L_1-norm penalty function acting on the location and the dispersion parameter. In order to limit the possible estimation bias introduced by the penalisation, we also implement an adaptive weighting rule that reduces the bias on informative variables.

We suggest a data resampling procedure to quantify the contribution of each variable to the clustering process. The advantage of such procedure is that we can explicitly measure the relative importance of each feature we retain in the model. This step enables us to rank the genes and the price dynamics statistics according to their selection probability and suggests an ordering criteria for a

more specific investigation of the informative variables. At the same time, the resampling procedure improves model selection by making the inferential process of discovering the true number of clusters more accurate.

As it is commonly done, given the complexity and high dimension of the mixture representation, the only viable approach to estimate the unknown parameters of the model is to implement an Expectation Maximization algorithm. In particular, we need to derive an algorithm that takes into account the penalty term. Moreover, the implementation code has to be optimized so that it can deal with data matrices of the dimensions of gene dataset which is in the region of ten of thousands.

The second model we introduce to perform supervised clustering is a mixture of Lasso regressions with t-errors. We consider the situation, such as is the case with the financial data problem, where at each sample we observe an array of explanatory variables and an associated response variable. The model is then supposed to explain as accurately as possible the vector of the dependent variables as a function of the design matrix.

We propose a model that expresses the dependent variable as a linear combination of the independent variables. The mixture structure we adopt, gives us the extra flexibility to fit an heterogeneous population where we presume that different regression curves might be needed to explain the different sub-populations. In our investigation, it allows us to fit, for each cluster of market, a separate linear combination of the relevant price dynamics statistics to explain the trading performance. Following a Bayesian approach, we specify a hierarchical representation of the mixture model where each response variable is marginally distributed as a Gaussian distribution with mean equal to the linear predictor.

To make sure the model is robust to outliers that otherwise would bias the estimation of the regression coefficients we introduce an auxiliary variable which allows the regression errors to be t-distributed. To further filter out any measurement noise that might be erroneously fitted by the regression curves, we specify a convenient hierarchy of priors and hyperpriors that deliver a Lasso type estimate of the regression coefficients. The effect of the regularization is to shrink the coefficients towards zero and therefore limit the impact of the noisy datapoints, which

otherwise would have too much weight especially in markets with a shorter history.

Since we also desire the clustering method to return a sparse solution, where only the truly informative variables are retained, we introduce a latent indicator vector that dictates which variables should be included and which variables should be excluded from the model. As we do not have any strong prior belief about which specific features of the price dynamics can influence the profitability of the trading strategy, we hope the model can automatically identify the most important ones.

Given the hierarchical structure of the priors and given the extra auxiliary variables we had to introduce in order to perform cluster assignment and variable selection, the model becomes too complex and high-dimensional for an explicit estimation of the unknown parameters. The only viable solution is to approximate the posterior distribution of the quantities of interest by devising an efficient sampling routine. We propose a Particle Markov chain Monte Carlo simulation procedure that alternates a conditional Sequential Monte Carlo algorithm to sample from the posterior of the clustering labels, with a Metropolised Gibbs sampler that updates the other relevant parameters conditional on the proposed cluster assignment.

Both supervised and unsupervised models are first tested on simulated data under multiple scenarios in order to verify that they possess the properties we require: accuracy, parsimony, sparsity and robustness. We then fit the models to the bioinformatics and financial datasets. The insightful and promising results that both models returns, confirm that they are appropriate for the problems we want to investigate.

1.3 Outline

The outline of the thesis is as follows.

In Chapter 2 we review the relevant literature discussing supervised and unsupervised learning approaches. We highlight the prominent role that mixture models have gained in recent years as an efficient and elegant method for clustering heterogeneous datasets. We illustrate how mixture models can be efficiently fitted in a maximum likelihood or Bayesian framework and discuss what solutions

have been explored to deliver a sparse solution.

In Chapter 3 we introduce a penalised mixture of Student's t distributions model in order to discover clusters that may exist in datasets that present noisy or extreme observations. We illustrate how we can achieve accurate variable selection by imposing an adaptive L_1-norm penalty function acting on the location and the dispersion parameter. A resampling procedure for model selection and variable ranking has also been proposed. To efficiently fit the the model we implement a modified EM algorithm that returns the maximum likelihood estimates of the unknown mixture parameters. The clustering and variable selection accuracy is assessed simulating several scenarios representative of real life situations.

In Chapter 4 we introduce, in a Bayesian framework, a mixture of Lasso regressions with t-errors. We illustrate how the hierarchical structure of priors we chose leads to a clustering model with the desired properties of robustness and sparsity. The parameters of interest are effectively estimated by implementing a simulation procedure that allow us to sample their posterior distribution. The performance and sensitivity of the model is then tested on experimental data.

In Chapter 5 we apply the penalised mixture of t distributions to cluster a cohort of patients diagnosed with breast cancer. The aim is to identify the subtypes of breast carcinoma that have clinical relevance and isolate the informative genes that show a distinct expression level pattern across clusters. By ranking the genes according to their importance we also hope to facilitate the biological interpretation of the results.

In Chapter 6 we implement both proposed models to find a reasonable partition of financial markets. We base the clustering algorithm on a collection of statistics that represent the main features of the price dynamics that characterize each market. The objective is to identify a more appropriate systematic trading strategy for each cluster and engineer an algorithm whose parameters can be calibrated on groups of similar markets.

In Chapter 7 we summarize the findings of the thesis and suggest promising routes for future work.

Chapter 2

Mixture Models and Variable Selection: A Review

2.1 Introduction

The challenge of studying real life datasets has raised the problem of formulating models able not only to deal with noisy observations and outliers but also to identify and filter out all non essential information. In this chapter we review the literature within supervised and unsupervised learning and give particular attention to models that demonstrate robustness and admit sparse solution. We highlight the properties of mixture models that make it a flexible and robust approach to discover hidden structure in the data. Both Maximum Likelihood (ML) and Bayesian approaches to statistical inference are discussed and both will be adopted throughout the thesis.

The outline of the chapter is as follows. In section 2.2 we introduce mixture models in their general notation as these will be the models we are going to develop and implement further in the thesis. In section 2.3 we discuss unsupervised learning. After reviewing other non-model based clustering algorithms, we give a more detailed interpretation of mixture models applied to cluster analysis. In section 2.4 we focus our attention on the variable selection problem and review different approaches proposed in the Maximum Likelihood and Bayesian framework

to solve it. In section 2.5 we discuss the role of linear models for regression and classification in supervised learning and how they can improve their flexibility and extend their reach in a mixture model framework. In section 2.6 we note how the variable selection problem in the context by linear regression models.

2.2 Mixture Models

With ever increasing computing power and data storage capabilities, very large scale scientific analyses are feasible in fields as diverse as physics, medicine or finance. Statistical analysis is essential to process all the recorded samples and machine learning methods can provide the tools to extrapolate information from large datasets. Mixture models in particular can uncover hidden structure in the data by identifying differences and similarities in distinct observations and clustering them.

Finite mixture models have been studied since at least the work of Newcomb (1886) and Pearson (1894), but only more recently their potential has been recognised and exploited. This started with the work of Wolfe (1970) accelerating following the books of Everitt and Hand (1981) and Titterington et al. (1985); McLachlan and Peel (2000).

The success of mixture models can be explained by the fact that they provide a flexible way to model complex probability distributions that would not be easily described by simpler pdfs. They also provide a natural framework for statistical modelling of heterogeneous population when data are thought to belong to one of several possible classes, but whose individual class memberships are unavailable.

In its most general formulation, a mixture of distributions is just a convex combination of other distributions. Given a vector of observations $y_i \in \mathbb{R}^p$ for $i = 1, \ldots, n$, the probability distribution of $\boldsymbol{y} = (y_1, \ldots, y_n)$ is said to follow a mixture model if the density $p(\cdot)$ is:

$$p(\boldsymbol{y}) = \sum_{k=1}^{K} w_k \, f_k(\boldsymbol{y}) \tag{2.1}$$

where $f_k(\cdot)$ for $k = 1, \ldots, K$, with $K \in \{1, \ldots, n\}$, are the component densities defined on the real space such that $\int_{\mathbb{R}^p} f_k(u)du = 1$. Note that even if we will only consider parametric type densities, mixture models are flexible enough to also cope with non-parametric functions. The non-negative mixing proportions $\boldsymbol{w} = (w_k, \ldots, w_K)$ satisfy the condition $\sum_{k=1}^{K} w_k = 1$ for $0 \leq w_k \leq 1$.

By comprising either finite or infinite number of components, not necessarily of the same family, mixture models can describe different features of the data. This property makes them particularly suitable for density estimation and cluster analysis.

For density estimation, mixture models provide a convenient semi-parametric framework to approximate arbitrarily well any continuous distribution. By using a sufficient number of Gaussians, and by adjusting their means and covariances as well as their weights, almost any continuous density can be approximated with arbitrary accuracy, see Escobar and West (1995); Roeder and Wasserman (1997); Bishop (2007). For example, by letting the number of components grow to $K = n$, the mixture becomes a nonparametric kernel estimator of the density, (McLachlan and Peel, 2000).

Beside providing a framework for building more complex probability distributions, finite mixture models are used to model clusters of data. The key intuition is that since the expressed features of samples from the same cluster are similar but not identical, it is reasonable to assume the existence of a probability distribution of features for each component. Samples from different clusters, on the other hand, should be characterized by different features. It is then natural to represent the combined population taken from all clusters as a mixture of distributions.

In light of this physical interpretation of mixture models, they have been extensively applied to solve *clustering* problems, which comprise estimating the parameters of the individual scaled components, estimating the classification probabilities of the observed or future data points or simply assigning each observation to a cluster of similar samples. These clustering properties make finite mixtures our model of choice to investigate the real life problems we described in chapter 1.

2.3 Unsupervised Learning

Unsupervised learning refers to the problem of trying to find hidden structure in unlabelled data where there is no error or target values that can guide and assess the learning process. This is the case of human breast cancer dataset we are going to study in chapter 5, where we only have the gene expression level for several patients and no information is available concerning the membership of the samples to any predefined class. We specify unsupervised clustering to distinguish it from semi-supervised approach that makes use of a small amount of supervision, see Grira et al. (2004) and Zhu and Goldberg (2009) for a review. What we are most interested in is discovering groups of similar objects, e.g. similar gene expression profiles, and how finite mixture contribute to this process. We should first briefly mention other existing methods competing with mixture models to solve this problem.

Non-Model Based Algorithms

Cluster analysis broadly refers to all methods that aim to organise a collection of objects into non-overlapping clusters, such that items within a cluster are more similar to each other than they are to items of different clusters.

Not all clustering methods found in the literature need to assume a probabilistic structure about the data, as for mixture models. Non-model based algorithms, like partitioning and the hierarchical approach, are only required to choose a proper metric to measure the degree of dissimilarity of each pair of items and use this information to identify homogeneous clusters. The notion of similarity can be expressed in many different ways, depending on the domain-specific assumptions, for example Euclidean distance for interval scaled variables or simple matching coefficient for nominal variables (Kaufman and Rousseeuw, 2005).

Hierarchical algorithms can be traced back to the work of Ward (1963) and produce a hierarchical structure by progressively combining or dividing existing groups. Agglomerative methods start with as many clusters as the number of objects and then successively merges them until only one large cluster remains

which is the whole data set. Divisive methods on the other hand follow the opposite direction and start by considering the whole data set as one cluster and then split up clusters until each object form is own cluster.

Partitioning algorithms specify an initial number of groups and iteratively reallocate observations between them until some equilibrium is attained. All algorithms within this class abide to two requirements: that each group contains at least one object and that each object belongs exactly to one group. Among the best known partitioning algorithm is k-means, originally presented by MacQueen (1967); see Kaufman and Rousseeuw (2005) for a review of other popular methods .

2.3.1 Mixture Models for Cluster Analysis

In a more rigorous formulation, given n observations characterised by p features, $y_i \in \mathbb{R}^p$ for $i = 1, \ldots, n$, the goal of a clustering procedure is then to partition the n samples into coherent groups. Following (2.1), each of the K distinct clusters is represented by a different density function, $f_k(\cdot)$, for $k = 1, \ldots, K$ and the p-dimensional vector $\boldsymbol{y}_i$ is then just a random sample from a population with probability density function:

$$p(\boldsymbol{y}_i) = \sum_{k=1}^{K} w_k \, f_k(\boldsymbol{y}_i) \tag{2.2}$$

where $\boldsymbol{w} = (w_k, \ldots, w_K)$ are as before the non-negative mixing proportions and $\boldsymbol{y}_i$ is thus assumed distributed according to either of the K density functions f_k with probability w_k. Once the data have been observed the inferential goal is then to estimate the weights and the parameters of the components' densities and possibly also the number of groups K. Under the assumption that n samples are independent, the mixture model of the entire dataset yields a joint pdf for $\boldsymbol{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$:

$$p(\boldsymbol{y}) = \prod_{i=1}^{n} \sum_{k=1}^{K} w_k \, f_k(\boldsymbol{y}_i) \tag{2.3}$$

This formulation allows mixture models to explicitly account for the experimental noise frequently encountered when analysing real data (Smolkin and Ghosh, 2003). It is, in theory, flexible enough to be able to use any parametric density function $f_k(\boldsymbol{y}_i|\boldsymbol{\Theta}_k)$ to describe the component distribution. It would also be possible to adopt a non-parametric component density. However, by staying in the parametric framework, one is able to keep the estimation problem reasonably tractable.

Parametric Mixtures

Now we consider parametric mixtures. The joint density of $\boldsymbol{y}$ given parameters $\boldsymbol{\Psi}$ is

$$p(\boldsymbol{y}|\boldsymbol{\Psi}) = \prod_{i=1}^{n} \sum_{k=1}^{K} w_k \, f_k(\boldsymbol{y}_i|\boldsymbol{\Theta}_k) \tag{2.4}$$

For each cluster k we now have a set of unknown parameters $\boldsymbol{\Psi}_k = (w_k, \boldsymbol{\Theta}_k)$ which includes the mixing coefficient w_k and the parameter vector $\boldsymbol{\Theta}_k$ that characterises the density function of the component. Whilst any parametric density function could be considered, in several instances (Pearson, 1894; Day, 1969; Wolfe, 1970; McLachlan and Peel, 2000; He et al., 2006) it has been deemed satisfactory to assume Gaussian components. A noticeable aspect is that only the location and scale parameters need to be specified $\boldsymbol{\Theta}_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$. On the other hand this particular choice of probability distributions may not always fit adequately the data we want to model. Depending on the observed characteristics of the data, different alternative parametrisations have been considered such as skew normal distribution in Lin et al. (2007) to compensate for asymmetry in data or normal inverse Gaussian distribution in Karlis and Santourian (2009) to model skewed and fat-tailed observations. The most frequent drawback that Gaussian components still suffer is the lack of robustness in the presence of high measurement noise and outliers. It can lead to an inflated number of detected clusters since additional components are needed to capture the heavy tail distribution of some variables (Peel and McLachlan, 2000; Liu and Rattray, 2010).

The Student's t distribution, as will be adopted in chapter 3, has already been successfully used for robust model-based clustering in several studies involving gene expression data (Liu and Rubin, 1995; Jiao and Zhang, 2008; Jiao, 2010). The degrees of freedom parameter $\boldsymbol{\nu}$ controls the exponential tail decay of the Student's t density which can be slower than a Gaussian distribution and yield longer tails (Kotz and Nadarajah, 2004).

Even if estimating $\boldsymbol{\nu}$ can be particularly problematic especially in a Bayesian likelihood context, since it does not allow a closed form solution (Fernandez and Steel, 1999), it can adjust the weight of extreme observations, being either genuine outliers or sampling errors, therefore reducing the bias when estimating the location and dispersion parameters (Peel and McLachlan, 2000; McLachlan et al., 2002). Each of the K multivariate Student's t densities components is then fully specified by the set of unknown parameters $\boldsymbol{\Theta}_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\nu}_k\}$ and have to be inferred from the observed data $\boldsymbol{y}$ together with the number of clusters K and the mixing coefficients.

Irrespective of the particular choice of parametrization, the estimation challenge posed by finite mixture models has been the object of several studies since Pearson (1894) approached this problem using the methods of moments. With the support of increasing computational power Maximum Likelihood Estimation (MLE), via the Expectation Maximisation (EM) algorithm (Dempster et al., 1977), and the Bayesian approach via sampling based methods, e.g. Markov Chain Monte Carlo (MCMC) following the papers by Metropolis et al. (1953); Hastings (1970), have proved to be the most effective methods to estimate the parameters of a finite mixture model. Since these are the two approaches that we followed and implemented in the thesis, in the next sections we are going to give a general introduction and more detailed review of existing literature on likelihood and Bayesian inference applied to mixture models.

2.3.2 Likelihood-Based Approach

In the MLE approach, the best estimate of $\boldsymbol{\Psi}$ is the one that yields the highest value of (2.4) given $\boldsymbol{y}$. Note that for many parametric families $f_k(\cdot|\boldsymbol{\Theta})$ the likelihood function can become unbounded as, for example, would happen with a univariate Gaussian component if we set the location parameter equal to one observation and let the dispersion of that component tend to zero. This problem can be avoided by constraining the variances of all components to be equal even though this assumption is not always supported by evidence.

Notwithstanding this issue, in practise it is often easier to maximise the logarithm of the likelihood function, $L(\boldsymbol{\Psi})$, which allows a more tractable formulation substituting a product with a sum. In regular situations the MLE estimate $\hat{\boldsymbol{\Psi}}$ is then found by solving:

$$\partial \log L(\boldsymbol{\Psi})/\partial \boldsymbol{\Psi} = 0 \tag{2.5}$$

Unfortunately the derivative of the log-likelihood function with respect to mixing proportions $\boldsymbol{w}$ and density parameters $\boldsymbol{\Theta}$ is typically complicated and severely multi-modal, rarely lending itself to mathematical treatment and analytical closed form solutions or straightforward numerical optimisation. The standard procedure for finding the MLE in almost all cases is the Expectation Maximization algorithm introduced by Dempster et al. (1977) and described in details by McLachlan and Krishnan (2008).

Missing Data Formulation

The model can be rewritten in a missing data framework, which facilitates parameter estimation (McLachlan and Peel, 2000). While one would not directly observe from which component of the mixture $\boldsymbol{y}_i$ has been sampled from, it is useful for the inferential process to introduce a K-dimensional component indicator vector $\boldsymbol{z}_i = (z_{i,1}, \ldots, z_{i,K})$ to indicate its cluster membership, see Bishop (2007). The collection of latent variables $\boldsymbol{z} = (z_1, \ldots, z_n)$ together with the observed samples $\boldsymbol{y}$ are usually referred to as the complete data set. Each $\boldsymbol{z}_i$ is a binary variable

defined for each sample $\boldsymbol{y}_i$ as

$$z_{i,k} = \begin{cases} 1 & \text{if} \quad \boldsymbol{y}_i \quad \text{belongs to component} \quad k \quad \text{for} \quad k = 1, \ldots, K \\ 0 & \text{otherwise} \end{cases} \tag{2.6}$$

Thus $\boldsymbol{z}_i$ follows a multinomial distribution, $\boldsymbol{z}_i \sim \mathcal{M}(1|w_1, \ldots, w_K)$ which can be interpreted as one draw on K categories with probabilities equal to the mixing proportions $\boldsymbol{w}$:

$$p(\boldsymbol{z}_i) = w_1^{z_{i,1}} \, w_2^{z_{i,2}} \, \ldots \, (1 - w_1 - \ldots - w_{K-1})^{z_{i,K}} = \prod_{i=1}^{K} w_i^{z_{i,k}} \tag{2.7}$$

The conditional distribution of $\boldsymbol{y}_i$ given $z_{i,k} = 1$ is then the density function of the k component $f_k(\boldsymbol{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k)$ or more explicitly for a mixture of t distributions:

$$p(\boldsymbol{y}_i|\boldsymbol{z}_i) = \prod_{k=1}^{K} f_k(\boldsymbol{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k)^{z_{i,k}} \tag{2.8}$$

By the law of total probability, we know that the marginal distribution of $\boldsymbol{y}$ can be obtained by summing the joint distribution of $\boldsymbol{y}$ and $\boldsymbol{z}$ over all possible states of $\boldsymbol{z}$. We note then that maximizing $\log L(\boldsymbol{\Psi})$ is equivalent to maximizing the log-likelihood of the complete data set, $\log L_c(\boldsymbol{\Psi})$,which we can derive by combining (2.7) and (2.8). Under the assumption that the observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ are i.i.d. we obtain the explicit form:

$$\log p(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{\Psi}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{i,k} \, \{\log w_k + \log f_k(\boldsymbol{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k)\} \tag{2.9}$$

The logarithm now acts directly on the component density function allowing the log-likelihood of each component k to be maximised independently and contributing to the MLE only if the latent variable $z_{i,k} = 1$. Aside from the computational problems associated with avoiding singularities in the likelihood surface, there may be several reasonable local maxima which might give quite different estimates of $\boldsymbol{\Psi}$.

The complete likelihood function of the missing observations, namely the group identifiers, in conjunction with the observed data has a much more appealing form that can be exploited by the Expectation Maximization algorithm; we will discuss this in more detail in chapter 3. We now introduce the Bayesian approach to estimation of the mixture models and discuss some of the methods found in the literature.

2.3.3 Bayesian Approach

Whereas the MLE provides a point estimate of every parameter of the mixture model, Bayesian methods treat $\boldsymbol{\Psi}$ as unknown quantities about which probability statements can be made. Before observing the data $\boldsymbol{y}$, the uncertainty is expressed as a prior distribution which represents our knowledge or belief about the value of the parameters. Having sampled the data, the prior belief is reviewed and a posterior distribution $\pi(\boldsymbol{\Psi}|\boldsymbol{y})$ conditional on $\boldsymbol{y}$ is constructed. The essence of the Bayesian inferential process is described by

$$\pi(\boldsymbol{\Psi}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\Psi})\, p(\boldsymbol{\Psi})}{p(\boldsymbol{y})} \tag{2.10}$$

where $p(\boldsymbol{y}) = \int_{\boldsymbol{\Psi}} p(\boldsymbol{y}|\boldsymbol{\Psi})\, p(\boldsymbol{\Psi})\, d\boldsymbol{\Psi}$. Many quantities or probability distributions of interest can be written as posterior expectations

$$\mathbb{E}(\phi(\boldsymbol{\Psi})|\boldsymbol{y}) = \int \phi(\boldsymbol{\Psi})\, \pi(\boldsymbol{\Psi}|\boldsymbol{y})\, d\boldsymbol{\Psi} \tag{2.11}$$

where $\phi : \boldsymbol{\Psi} \to \mathbb{R}$.

This procedure is valid generally, but when we follow the Bayesian inferential process for mixture models, it is convenient, as we have seen for the MLE approach, to adopt a missing data formulation. In the Bayesian approach we follow in chapter 4, (see also Marin et al. (2005); Bishop (2007)), the missing allocation variable z_i with $z_i \in \{1, \ldots, K\}$ is now a categorical variable with probability mass function

$$p(z_i = k|\boldsymbol{w}, K) = w_k \tag{2.12}$$

Given the vector of missing labels $\boldsymbol{z} = (z_1, \ldots, z_n)$, Bayesian inference simply considers each possible allocation of the dataset, attaches a posterior probability to this allocation, and then constructs a posterior distribution for the parameter set $\boldsymbol{\Psi}$ conditional on this allocation

$$\pi(\boldsymbol{\Psi}|\boldsymbol{y}) = \frac{\sum_z L_c(\boldsymbol{\Psi})\, p(\boldsymbol{z}|\boldsymbol{\Psi})\, p(\boldsymbol{\Psi})}{\int \sum_z L_c(\boldsymbol{\Psi})\, p(\boldsymbol{z}|\boldsymbol{\Psi})\, p(\boldsymbol{\Psi}) d\boldsymbol{\Psi}} \tag{2.13}$$

where $p(\boldsymbol{z}|\boldsymbol{\Psi})$ is the conditional probability of $\boldsymbol{z}$ given $\boldsymbol{\Psi}$ and $L_c(\boldsymbol{\Psi})$ denotes the likelihood of the complete data set $\boldsymbol{y}_c = (\boldsymbol{y}, \boldsymbol{z})$.

To illustrate the concept we use K Gaussian components which are fully specified by the location and dispersion parameter $\boldsymbol{\Theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$. The posterior distribution of the entire set of unknown parameters for the mixture model, assuming we know K, is

$$\pi(\boldsymbol{w}, \boldsymbol{\Theta}|\boldsymbol{y}) \propto \left(\prod_{i=1}^{n} \sum_{k=1}^{K} w_k\, f_k(\boldsymbol{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) p(\boldsymbol{w}, \boldsymbol{\Theta}) \tag{2.14}$$

where the choice of priors on $\boldsymbol{w}$ and $\boldsymbol{\Theta}$ can be a contentious issue as noted by Aitkin (2001) who reviewed different options. In this thesis we will be using weakly informative priors about the mixture model parameters, including the number of components K, that are justified by our knowledge of the problems at hand gained from related studies, from alternative clustering methods or from relevant evidence.

As an illustrative example, Escobar and West (1995) assumed a mixtures of Dirichlet processes where the prior of mixing coefficients is a symmetric Dirichlet distribution:

$$\boldsymbol{w} \sim \mathcal{D}ir(\boldsymbol{w}|\delta) \tag{2.15}$$

and the symmetry is forced by fixing the same parameter δ for all components. The prior on the location parameter is a Normal distribution

$$\boldsymbol{\mu}_k \sim \mathcal{N}(m_k, \boldsymbol{\Sigma}_k/(v_k)) \tag{2.16}$$

and for the covariance matrix an Inverse Wishart distribution

$$\boldsymbol{\Sigma}_k \sim \mathcal{IW}(a_k, b_k) \tag{2.17}$$

where m_k, v_k, a_k, b_k are the hyperparameters necessary to specify the first level priors and can themselves be object of second level priors.

In particular, as part of the cluster analysis, we are interested in the classification probability of the observed data point $\boldsymbol{y}_i$ which is given by

$$\pi(z_i = k|\boldsymbol{y}) = \int p(z_i = k|\boldsymbol{y}, \boldsymbol{w}, \boldsymbol{\Theta})\, \pi(\boldsymbol{w}, \boldsymbol{\Theta}|\boldsymbol{y})\, d\boldsymbol{w}\, d\boldsymbol{\Theta} \tag{2.18}$$

Unfortunately (2.18) requires one to analytically calculate an intractable integral. For the mixture models we have just illustrated, it is unfeasible to evaluate the posterior distribution $\pi(\boldsymbol{\Psi}|\boldsymbol{y})$ or indeed to compute expectations of any function with respect to this distribution. The dimensionality of the latent space can become too high to work with analytically and the form of the posterior distribution can be too complex, even for a convenient choice of prior, to allow expectations to be computed explicitly. It is not surprising then that Bayesian inference on mixture models has become a viable approach only following the development of stochastic numerical approximation techniques such as Monte Carlo methods.

Identifiability

While we will adopt Bayesian mixture model especially for supervised learning, we should be warned here that a Bayesian analysis present difficulties with the general mixture model representation at both the exploration stage and the interpretation stage, see Titterington et al. (1985); Celeux et al. (2000). Parameters can be non identifiable since the likelihood

$$L(\boldsymbol{w}, \boldsymbol{\Theta}) = \prod_{i=1}^{n} [w_1 f(\boldsymbol{y}_i|\boldsymbol{\Theta}_1) + \cdots + w_K f(\boldsymbol{y}_i|\boldsymbol{\Theta}_K)] \tag{2.19}$$

is symmetric in the components $1, \ldots, K$, which means the likelihood is the same for all permutations of the parameters.

If the prior distribution of the parameters is also invariant under permutations of the parameters, as it would happen if we had no real prior information about the components, then the posterior distribution will be similarly invariant, resulting identical for each mixture component and showing up to $K!$ modes, (Celeux et al., 2000; Jasra et al., 2005).

Whilst this is usually not a problem for MLE via EM algorithm (McLachlan and Peel, 2000), the switching of components labels compromises any inference drawn from numeric approximation methods, such as the Monte Carlo method we introduce next, which are based on ergodic average over samples from the posterior distribution.

While an ordering constraint on mixing proportions or other component parameters can be imposed ex post, after the simulations have been completed, e.g. Richardson and Green (1997), this approach does not always work and different solutions like alternative identifiability constraints, relabelling algorithms and label invariant loss functions have been proposed, see Jasra et al. (2005).

The label switching problem is less threatening in our study since we can use prior knowledge derived from MLE results to guide the relabelling algorithm after the simulation has run.

Monte Carlo Methods

Let $\pi(\cdot)$ be a probability density on $\mathcal{X}$, we want to calculate the expectation w.r.t $\pi(\cdot)$ of any suitably $\pi-$integrable function $h(\cdot)$ defined on $\mathcal{X}$ such that $h : \mathcal{X} \to \mathbb{R}$. The expectation is given by the integral

$$I(h) = \int h(x)\, \pi(x) dx \tag{2.20}$$

Since it is often too complex to exactly evaluate (2.20) using analytical techniques, we need to resort to approximation schemes either deterministic or stochastic in nature.

Deterministic algorithms are based on analytical approximations of the posterior distribution. Therefore, the more complex the model becomes the higher the need of simplifying assumption which will lead us further away from the exact results, see Roberts et al. (1998); Figueiredo and Jain (2002).

Alternatively, the integral (2.20) can be approximated through stochastic techniques such as Monte Carlo methods. This approach originated in the Los Alamos laboratories and was proposed to answer several mathematical problems encountered by scientists while trying to build the atomic bomb, see Robert and Casella (2011) for a historical account. The basic premise of Monte Carlo methods is to perform approximations through the random generation of variates, ensuring that such variates are distributed according to $\pi(\cdot)$. The *perfect* algorithm would sample directly from the target distribution of interest, $\pi(\cdot)$; generate N values $x_1, \ldots, x_N$ all i.i.d. $\sim \pi(\cdot)$ to obtain the empirical estimate

$$\hat{\pi}(x) = \frac{1}{N} \sum_{j=1}^{N} \delta_{X^j}(x)$$

where $\delta_{x^0}(x)$ is the Dirac delta mass at x^0. The integral (2.20) is then approximated by

$$I^{MC}(h(x)) = \int h(x)\, \hat{\pi}(x) dx = \frac{1}{N} \sum_{j=1}^{N} h(X^j) \tag{2.21}$$

The main advantage of Monte Carlo methods over standard approximation techniques is that the variance of the approximation error decreases at a rate of $\mathcal{O}(N^{-1})$ regardless of the dimension of the space $\mathcal{X}$.

Importance Sampling

In most practical situations, when the posterior $\pi(\mathbf{\Psi})$ is complex and high-dimensional, we might find it unfeasible to sample directly from a target distribution. In addition, especially in Bayesian statistics, the target distribution $\pi(x)$ can often only

be evaluated up to a normalization constant

$$\pi(x) = \frac{\tilde{p}(x)}{Z_p}$$

where $\tilde{p} : \mathcal{X} \to \mathbb{R}^+$ is known pointwise and the normalising constant

$$Z_p = \int \tilde{p}(x)\, dx$$

might be unknown.

Importance Sampling (IS) (Marshall, 1956; Geweke, 1989; Robert and Casella, 2004), overcomes the problem of having to generate a set of i.i.d. values from $\pi(x)$ by drawing instead from an easier to sample *importance density* $q(x) = \tilde{q}(x)/Z_q$ and weighting the samples so that they approximate the empirical measure $\tilde{p}(x)$

As long as the support of $q(x)$ covers the support of $\pi(x)$:

$$\pi(x) > 0 \Rightarrow q(x) > 0$$

by the Radon-Nykodym theorem, see Jacod and Protter (2004), the integral (2.20) can be expressed in the form

$$I(h) = \int h(x)\, \frac{\pi(x)}{q(x)}\, q(x)\, dx$$

and the following identities hold true

$$\pi(x) = \frac{\omega(x)\, q(x)}{Z_p},$$

$$Z_p = \int \omega(x)\, q(x)\, dx$$

where the unnormalized weight function $\omega(x)$ is the ratio of the target distribution $\tilde{p}(x)$ with respect to importance distribution $q(x)$

$$\omega(x) = \frac{\tilde{p}(x)}{q(x)}$$

Having drawn N i.i.d. samples from $q(x)$ we can compute the empirical measure and estimate the value of the normalising constant

$$\hat{\pi}(x) = \sum_{j=1}^{N} W^j \, \delta_{X^j}(x)$$

$$\hat{Z}_p = \sum_{j=1}^{N} \omega(X^j)$$

where W^j are the normalised weights

$$W^j = \frac{\omega(X^j)}{\sum_{l=1}^{N} \omega(X^l)}.$$

Therefore, when we can not have a perfect Monte Carlo sampling, we can still estimate the classification probabilities in (2.18) using the importance sampling approximation

$$I^{IS}(h(x)) = \int h(x) \, \hat{p}(x) dx = \sum_{j=1}^{N} W^j \, h(X^j)$$

$I^{IS}(h(x))$ is an asymptotically consistent estimator of $I(h(x))$, the difference is that, unlike $I^{MC}(h(x))$, is biased for finite N unless the normalising constant is known analytically in which case we can produce an unbiased estimator.

Given a function $h(\cdot)$ it would be relatively easy to derive $q(x)$ such that the asymptotic variance of $I^{IS}(h(x))$ is minimised, but one cannot typically sample from it. For importance sampling to perform well in practice, the sampling distribution $q(x)$ should not be small in regions where $p(x)$ may be significant. The risk otherwise is that $h(x)p(x)$ might have a significant proportion of its mass concentrated over small regions of the $\mathcal{X}$ space with the consequence that few importance weights $\{W^j\}$ will dominate all the others. This problem can not be solved by increasing or reducing the number of samples N. Intuitively it seems sensible rather to choose the proposal density q to be as close as possible to the target $\tilde{p}$ such that the variance of the importance weights or equivalently the variance of $\hat{Z}_p$ is

minimised:

$$\int \left(\frac{\tilde{p}(x)}{q(x)} - 1 \right)^2 q(x)\, dx.$$

The limit of such approach is that if q is very close to $\tilde{p}$ it might also be equally difficult to sample from, defying the purpose of the importance sampling.

Unfortunately, the optimal importance density is often unknown and, as the system dimension become very large, IS may fail because the weights collapse, as discussed by (Bickel et al., 2008; Bengtsson et al., 2008).

In similar situations a viable alternative is provided by the Markov Chain Monte Carlo (MCMC) methods which scales well with the dimensionality of the sample space, see Gelfand and Smith (1990). Via the construction of an ergodic Markov Chain that is irreducible and associated with an invariant probability distribution p, MCMC methods generate a random sample $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N$ suitable for simulation purposes, from the approximation of the integrals under p to the exploration of the support of p. We will refer to MCMC methods in chapter 4 and we defer till then to give a more detailed review of how it can be implemented to conduct inference on mixture models.

2.4 Variable Selection in Unsupervised Learning

In the previous sections we have introduced the general settings for solving clustering problems with mixture models. Here we focus our attention on a practical problem associated to studying high-dimensional datasets as the gene and financial datasets we want to investigate in this thesis. The aim is to look at existing methods that can help us identify the truly informative variables and improve the accuracy of the results.

The development and refinement of efficient ML and Bayesian estimation algorithms has contributed decisively to the success of finite mixture models of recent years. Nonetheless, despite their flexibility and model parsimony, high-dimensional datasets still pose a particularly difficult clustering problem to solve especially when not all the features recorded are useful to group observations correctly.

The datasets we want to investigate, are characterized by a number of observations which is a lot smaller than the number of variables, $n << p$. While our main goal is to cluster the n objects into K homogeneous groups, we are also interested in identifying the $1 \leq m \leq p$ features that are really informative for the clustering process. Being able to select only the m variables that show a significantly different distribution across clusters means that the remaining $q = p - m$ variables can be excluded from the model. An effective variable selection procedure can significantly improve the clustering accuracy of the algorithm which could otherwise be mislead by the noise introduced by the non informative variables.

At the same time, a clustering algorithm that delivers a parsimonious solution that only uses few discriminant variables will facilitate the interpretation of the results. Having fewer variables in the model make it easier to see the responsibility of a specific feature in characterizing certain clusters and help us focus our research. This process would be more straightforward in a supervised framework where we can infer from the loading factors which variables are more relevant.

To better appreciate the importance of variable selection in unsupervised clustering, consider the case of DNA microarray studies, (Eisen et al., 1998; Golub et al., 1999). The typical DNA microarray dataset has only a limited number of observations, say few hundreds, while the number of genes can be in the order of tens of thousands. The ideal unsupervised learning procedure then should help us to identify the different cluster of patients and simultaneously isolate the genes responsible. If we note an overexpression of a particular gene in a specific group of patients, we can conjecture that that gene might be linked to the insurgency of whatever pathology is present in that group.

Variable Selection in Non-Model Based Methods

Before discussing all the different approaches to variable selection via mixture models we review some significant variable selection approaches implemented in non-model based clustering methods. One popular method is to associate each variable with a binary variable that indicates its inclusion or exclusion and have

an ad hoc algorithm to search through the space of all possible combinations for the best sparse solution. Since the size of this space is exponential in the number of features, it is usually impractical to explore all combinations and so heuristic non-exhaustive methods have to be adopted. The drawback is that one generally loses any guarantee of finally identifying the correct subset of informative variables, assuming one exists at some point. To minimize this risk Fowlkes et al. (1988) and Fraiman et al. (2006) proposed a forward selection algorithm for sparse hierarchical and partitioning clustering respectively. Other top-down and bottom-up hierarchical approaches to uncover clusters in multiple, possibly overlapping subspaces of the multidimensional dataset, are reviewed by Parsons et al. (2004).

In a different approach, described by McLachlan et al. (2002), univariate models are fitted on each variable and only a small subset of variables that pass some significance threshold are retained. One noticeable shortcoming of this method is that it does not assess the joint effect of multiple variables. As a result, it might end up throwing away potentially valuable features, which are not useful for clustering individually but may yield an improvement when considered in conjunction with others.

Variable selection can also be achieved by assigning, to each feature, a weight which represents the value of the information it conveys towards the identification of clusters. Gnanadesikan et al. (1995); Friedman and Meulman (2004); Steinley and Brusco (2008) discuss several different weighting rules all based on a distance metric, while Witten and Tibshirani (2010) more recently refined a method that yields a sparse solution for k-means partitioning.

Other authors, like Fodor (2002); Ghosh and Chinnaiyan (2002), have devised different dimension reduction techniques that perform a transformation of the data and project them on a lower dimensional space. Principal component analysis, for example, will use the Euclidean distance between features profiles to produce linear combinations of the variables and so reduce the dimensionality of the problem. The main drawback is that the results are often difficult to interpret except when most of the coefficients of the linear combination are negligent.

2.4.1 Penalised Mixture Models

Within the mixture model literature, most of the research on feature selection pertains to supervised learning, both from a classification and regression point of view. Only more recently there has been an increasing interest from an unsupervised learning perspective. Here we focus our attention on the Maximum Penalised Likelihood Estimation (MPLE) which we will extend in chapter 3 to a robust class of mixture components. We then give a more detailed review of some significant examples and more recent contributions on this subject.

The fundamental intuition justifying variable selection in mixture models is that only the informative features are expected to show a markedly distinct distribution in a different cluster. Whereas, noise variables are represented as being sampled from one single common distribution that would not suggest any partitioning of the data. In the maximum likelihood estimation framework, model fitting and variable selection are realised simultaneously by imposing a convenient penalisation on the likelihood function. The approach proposed for mixture models by Pan and Shen (2007); Xie et al. (2008a); Wang and Zhu (2008) follows and adapts the original idea described by Tibshirani (1996); Fan and Li (2001) who used it in supervised learning to regularize regression's coefficients.

The key intuition is that the added penalty term forces the MLE estimates to shrink the component density relevant parameters and the variables whose parameters' values fall below a chosen threshold are considered non informative. In practice, a variable can be excluded from the model without prejudice when its parameters estimated values are the same across all clusters such that it does not convey any useful information to discriminate between clusters.

Generalising the different versions found in literature the penalised log-likelihood function is

$$\log L_p(\boldsymbol{\Psi}) = \log L(\boldsymbol{\Psi}) - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta})$$

which, expanding for the K components, yield

$$\log L_p(\boldsymbol{\Psi}) = \sum_{j=1}^{n} \left[\log \left(\sum_{k=1}^{K} w_k \, f_k(\boldsymbol{y}_j | \boldsymbol{\Theta}_k) \right) \right] - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_k) \tag{2.22}$$

where the penalty term $h_\lambda(\cdot)$ is a function of the relevant components density parameters and depends on the penalisation factor $\boldsymbol{\lambda}$. Besides being very effective in achieving a sparse solution, the formulation of (2.22) naturally fits within the EM framework without compromising the performance of the algorithm.

Penalised Gaussian Mixture Models

In the literature discussing the penalised likelihood approach applied to model-based cluster analysis, Gaussians mixture models have attracted most of the attention. It is thus not surprising that the most recent studies use this family of distributions to illustrate their proposed contributions to this area of research.

In their article Pan and Shen (2007) assume a Gaussian mixture model and introduce an $\mathbb{L}_1-$norm penalty function to exclude non-informative variables by regularising the location parameter of each component:

$$h_\lambda(\boldsymbol{\mu}) = \lambda \sum_{k=1}^{K} \sum_{d=1}^{p} |\mu_{k,d}|$$

The penalty grows proportionally to the product of the regularisation parameter λ and $|\boldsymbol{\mu}|$. Assuming the data has been previously centered, the term $-h_\lambda(\boldsymbol{\mu})$ forces the MLE process to shrink the estimated location parameter towards the common mean 0. An estimate different from zero would only be justified by a more than proportional increase in the log-likelihood term $\log L(\boldsymbol{\Psi})$. Those variables whose location parameter do not resist the shrinkage and collapse to 0 across all components, will not be relevant to identify the clusters and are effectively excluded from the model.

The choice of an appropriate value for λ is critical since it controls the amount of the shrinkage and ultimately sets the threshold below which variables are filtered

out. A common heuristic approach fits a series of models for different values of λ and then chose the best model according to the Bayesian Information Criterion (BIC), Schwarz (1978).

Adaptive Penalty Function

As a result of the shrinkage forced by λ, it has been noted that regularisation introduces an estimation bias of the parameters even for genuinely informative variables. Confronted with a similar problem in fitting regressions, Zou (2006) proposed an adaptive version of the standard LASSO originally presented by Tibshirani (1996). The strategy consists in imposing less penalisation on large coefficients using adaptive weights. Wang and Zhu (2008); Xie et al. (2008a) have followed this solution and suggested an adaptive version of the penalty function applied to mixture models:

$$h_\lambda(\boldsymbol{\Theta}) = \lambda \sum_{k=1}^{K} \sum_{d=1}^{p} \omega_{k,d} \left|\mu_{k,d}\right|$$

where the penalty weights $\boldsymbol{\omega}$ scale with the importance of the variable. The general rule adopted makes the weights inversely proportional to the absolute value of location parameter $\omega = 1/|\mu_{k,d}|^v$, where $v > 0$ modulates how quickly the shrinkage is removed as $\mu_{k,d}$ gets larger.

$\mathbb{L}_\infty-$Penalty Function

Instead of dealing with each of the cluster specific means individually, Wang and Zhu (2008) rather propose an $\mathbb{L}_\infty$-norm penalty function that treats $\mu_{k,d}$ for $k = 1, \ldots, K$ as interdependent groups, since they pertain to the same variable d. The penalty function they implement is

$$h_\lambda(\boldsymbol{\Theta}) = \lambda \sum_{d=1}^{p} w_d \max_k(|\mu_{1,d}|, \ldots, |\mu_{k,d}|, \ldots, |\mu_{K,d}|) \tag{2.23}$$

where $\max(|\mu_{1,d}|, \ldots, |\mu_{K,d}|) = \|(\mu_{1,d}, \ldots, \mu_{K,d})\|_\infty$ and ω_d is the adaptive weight $w_d = 1/\max_k |\mu_{k,d}|^v$. The $\mathbb{L}_\infty$-norm penalty guarantees that the estimated location

parameters of d will be zero for all clusters simultaneously if the maximum absolute value of $\mu_{k,d}$ for $k = 1, \ldots, K$ is shrunken to zero. When that happens the variable is excluded.

Grouped Penalization

The merits of grouping together multiple parameters arising from the same variable before imposing penalisation have also been recognised by Xie et al. (2008b) who proposed a penalty function that explicitly encourages all of the $\mu_{1,d}, \ldots, \mu_{K,d}$ to be exactly 0 at the same time. The approach described there is general, but assuming for illustration purposes an $\mathbb{L}_2$-norm penalty function we have:

$$h_\lambda(\Theta) = \lambda \sqrt{K} \sum_{d=1}^{p} \|\mu_{\cdot,d}\| \tag{2.24}$$

with $\|\mu_{.d}\| = \sqrt{\sum_{k=1}^{K} \mu_{k,d}^2}$. When most of the clusters' estimated location parameter appear to be close to zero, the quadratic mean will be less than a certain threshold and all $\mu_{k,d}$ for $k = 1, \ldots, K$ will be set equal to 0 thus making the variable d uninformative. In the same paper Xie et al. (2008b) also discuss an horizontal grouping where multiple variables are aggregated together based on any previous knowledge that might justify their joint inclusion or exclusion from the model. Assuming the p variables and their corresponding location parameters have been partitioned in M groups of possibly different dimension $\dim(\mu_i^m) = p_m$ with $\sum_{m=1}^{M} p_m = p$, the grouped horizontal penalty function suggested by the authors is:

$$h_\lambda(\Theta) = \lambda \sum_{k=1}^{K} \sum_{m=1}^{M} \sqrt{p_m} \, \|\mu_k^m\| \tag{2.25}$$

Hence if the weighted quadratic mean of the estimated location parameters of the variables in group m is small enough all elements of μ_k^m will be shrunken to exactly zero.

Variance Penalisation

In a different paper Xie et al. (2008a) followed the approach suggested by Pan and Shen (2007) and extended regularisation to the dispersion parameter. For a mixture of Gaussian components $\boldsymbol{\Theta}_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$, assuming features are independently expressed, i.e. a diagonal covariance matrix $\boldsymbol{\Sigma}_k = diag(\sigma^2_{k,1}, \ldots, \sigma^2_{k,p})$, they proposed two schemes:

$$\text{Scheme I:} \quad h_\lambda(\boldsymbol{\Theta}) = \lambda \sum_{k=1}^{K} \sum_{d=1}^{p} |\sigma^2_{k,d} - 1| \tag{2.26}$$

$$\text{Scheme II:} \quad h_\lambda(\boldsymbol{\Theta}) = \lambda \sum_{k=1}^{K} \sum_{d=1}^{p} |\log \sigma^2_{k,d}| \tag{2.27}$$

Except for the linear or logarithm transformation of the estimated standard deviation parameter, the schemes are equivalent as they both apply a L_1-norm penalty function that forces $\sigma_{k,d}$ to shrink towards 1. The intuition again is that once the data have been standardised, setting $\sigma_{k,d}$ equal to 1 for all components $k = 1, \ldots, K$ implies that the variable d irrelevant for clustering.

Pairwise Penalisation

The procedures seen so far select a variable if it is informative to separate at least one pair of clusters and remove it only when it has the same distribution across all clusters. Guo et al. (2010) discuss a method to identify which variables are discriminative for which pairs of clusters. They propose a pairwise fusion penalty function that penalises the distance between every pair of cluster centers. The shrinkage in this case pushes the centroids of non-separable clusters towards each other until eventually they collapse to the same value. In fact the pairwise penalty function is

$$h_\lambda(\Theta) = \lambda \sum_{d=1}^{p} \sum_{1 \le k < k' \le K} \omega^{(d)}_{k,k'} \, |\mu_{k,d} - \mu_{k',d}|$$

where $\omega_{k,k'}^{(d)} = |\tilde{\mu}_{k,d} - \tilde{\mu}_{k',d}|^{-1}$ is the adaptive weight with $\tilde{\mu}_{k,d}$ being the estimate of $\mu_{k,d}$ without penalisation. The data do not need to be centered and it is the difference between means that gets shrunk towards zero, not the means themselves. If $\tilde{\mu}_{k,d} = \tilde{\mu}_{k',d}$ then the variable d is not considered to be informative for separating cluster k and cluster k' but it might still be included in the model if it is informative to separate other clusters.

2.4.2 Bayesian Variable selection

Compared to Maximum Penalised Likelihood Estimation the Bayesian implementation of feature selection is more flexible but computationally more demanding. Even in the presence of high-dimensional datasets Bayesian methods can naturally address the problem of simultaneously selecting informative variables, uncovering the cluster structure of the observations and propose class prediction for future observations.

The Bayesian paradigm offers a coherent framework to overcome the major shortcomings of non-model based clustering algorithms like the k-means approach, (Bishop, 2007). Typically partitioning or hierarchical methods use greedy deterministic search procedures, which can be stuck at local minimum, and presume the existence of a single best subset of clustering variables. In practice, however, there may be several equally good subsets that define the true cluster structure. The Bayesian approach on the other hand offers a probabilistic criterion for assessing the uncertainty associated with model estimation and variable selection.

Principal Component Analysis

Liu et al. (2003) were among the first authors that were confronted with this problem and tried a Bayesian approach to standard dimension reduction. The authors proposed to perform a preliminary principal component analysis (PCA) or correspondence analysis (CA) to reduce dimensions, and then fit Gaussian mixtures to the data projected to the several major PCA or CA directions. The method combines the reparametrisation of the covariance matrix described by Banfield

and Raftery (1993) with a formal Bayesian modelling. The authors implement a Metropolis-Hastings algorithm to sample over the discrete space of possible subsets of the principal components. The advantage of this type of full Bayesian models is its ability to treat all involved variables in a coherent framework, to combine different sources of information, and to reveal subtle patterns by properly averaging out noise.

Sparsity Inducing Priors

To handle the problem of selecting only the informative features among the prohibitively vast number of variable subsets, an often adopted approach has been to introduce a latent p-dimensional binary vector $\boldsymbol{\gamma} = (0,1)^p$. Each element γ_d for $d = 1, \ldots, p$ indicates whether the d^{th} variable is informative in which case it assumes a value 1 or 0 otherwise. By explicitly modelling $\boldsymbol{\gamma}$ it is possible to make informed inference on joint and marginal posterior distributions of the variables. A suitable prior for the selection indicator is the Bernoulli distribution which, under the assumption that the elements of $\boldsymbol{\gamma}$ are independent, yields

$$p(\boldsymbol{\gamma}) = \prod_{d=1}^{p} \phi^{\gamma_d} (1-\phi)^{1-\gamma_d}$$

where the hyperparameter ϕ can be interpreted as the proportion of variables expected a priori to be informative.

Based on the binary latent indicator several models have been considered for their different advantages. Using a Gaussian mixture model, Tadesse et al. (2005) explored the idea presented by Law et al. (2004) and proposed a diagonal covariance matrix which implies each variable is independent from the others. In line with the key intuition exploited by the MPLE, all the irrelevant variables are thought to have been sampled from a common distribution and show similar expression patterns regardless of cluster membership. Assuming that $I(\gamma')$ is the index of the variables identified as informative, i.e. that should be included in the model, with $I(\gamma'')$ indicating the complementary set of variables that favour a

single multivariate normal density, the likelihood function is then rewritten as

$$p(\boldsymbol{y}_i|\boldsymbol{z}_i = k) \;=\; \mathcal{N}(\boldsymbol{y}_{i\,I(\gamma')}|\boldsymbol{\mu}_{k\,I(\gamma')}, \boldsymbol{\Sigma}_{k\,I(\gamma')})\,\mathcal{N}(\boldsymbol{y}_{i\,I(\gamma'')}|\boldsymbol{\mu}_{k\,I(\gamma'')}, \boldsymbol{\Sigma}_{k\,I(\gamma'')}).$$

A later study by Raftery and Dean (2006) tried to relax the assumption of complete independence between informative and non informative variables and assumed that the irrelevant variables could still be regressed on the relevant variables. Their modelling enforced the dependency link between the two types of variables. In this case three sets of variables are defined: $I(\gamma')$ the set of variables already selected, $I(\gamma'')$ the set of variables to be evaluated for inclusion or exclusion and $I(\gamma''')$ the remaining excluded variables. The hierarchical representation of the likelihood yields:

$$p(\boldsymbol{y}_i|\boldsymbol{z}_i = k) \;=\; p(\boldsymbol{y}_{i\,I(\gamma')}|\boldsymbol{z}_i)\,p(\boldsymbol{y}_{i\,I(\gamma'')}|\boldsymbol{y}_{i\,I(\gamma')})\,p(\boldsymbol{y}_{i\,I(\gamma''')}|\boldsymbol{y}_{i\,I(\gamma')},\, \boldsymbol{y}_{i\,I(\gamma'')}).$$

Following the same route Maugis et al. (2009) suggested a further improvement on the model and allow the irrelevant variables to be explained by only some of the relevant variables. The reviewed model accounts for the possibility that some of the uninformative features are independent of all the relevant variables while the remaining features are linked by some extent to some of the relevant variables. The set of informative variables is denoted by $I(\gamma')$ while its complement set contains the irrelevant variables and can be split into subset $I(\gamma'')$, comprising the redundant variables that can be explained according to a linear regression on $I(\gamma'^{*}) \subset I(\gamma')$, and into subset $I(\gamma''')$ comprising the remaining non informative and independent variables.

$$p(\boldsymbol{y}_i|\boldsymbol{z}_i = k) \;=\; p(\boldsymbol{y}_{i\,I(\gamma')}|\boldsymbol{z}_i)\,p(\boldsymbol{y}_{i\,I(\gamma'')}|\boldsymbol{y}_{i\,I(\gamma'^{*})})\,\mathcal{N}(\boldsymbol{y}_{i\,I(\gamma''')}|\boldsymbol{\mu}_{k\,I(\gamma''')}, \boldsymbol{\Sigma}_{k\,I(\gamma''')}).$$

The estimated distance between clusters' means, in the spirit of LASSO penalisation, is the focus of the study by Yau and Holmes (2011). A hierarchical Bayesian nonparametric mixture model is proposed to deal with situations where there is uncertainty about the clustering relevance of the variables. The hierarchi-

cal structure provides a flexible framework that can accommodate the hypothesis that some variable are informative only to separate specific clusters, while the non-parametric prior allows us to treat the number of mixtures as unknown. In order to induce sparsity, it is assumed that the standardised distance between any pair of clusters k and k' should have a Gaussian prior

$$\frac{(\mu_{k,d} - \mu_{k',d})}{\sqrt{2\sigma_d^2}} \sim \mathcal{N}(0, \lambda_d) \tag{2.28}$$

where $\boldsymbol{\lambda}$ is the shrinkage parameter vector that forces the cluster means towards a common location in a way that encourages sparsity. The relative importance and contribution towards clustering of each covariate can be gauged from the posterior distribution of $p(\lambda_d|\boldsymbol{y})$.

Model Selection

All methods we have just mentioned refer to the latent binary indicator vector $\boldsymbol{\gamma}$ to recast the variable selection problem as one of model selection and addresses it using approximate Bayes factors. In Raftery and Dean (2006), for example, the two models compared are: the one that assumes that the set of variables $I(\gamma'')$ does not provide further useful information once $I(\gamma')$ has been observed, and the other competing model that assumes $I(\gamma'')$ does add more useful information for the identification of the clusters.

The criteria followed to choose among the competing models consist in favouring the one that maximises the ratio of posterior to prior odds. When it is difficult to analytically evaluate the integrated likelihoods, the Bayes factors are approximated by the difference between the Bayesian Information Criteria BIC of the two models, see Schwarz (1978).

Subspace Clustering

Following a slightly different approach Hoff (2006) tried to reformulate, in a Bayesian framework, the subspace clustering method suggested by Friedman and Meulman

(2004). The main motivation for the model-based subspace clustering method is that only a small number of the attributes differentiate groups of observations, and among these attributes, only some will differ between any two particular clusters. The cluster specific means, $\boldsymbol{\mu}_k$, are modelled as a shift from the population wide mean $\boldsymbol{\mu}_k = \boldsymbol{\mu} + \boldsymbol{r}_k \times \boldsymbol{\delta}$ with $\boldsymbol{r}_k \in \{0,1\}^p$, $\boldsymbol{\delta}_k \in \mathbb{R}^p$ and the distribution of the shifts $\{(\boldsymbol{r}_i, \boldsymbol{\delta}_i) : i = 1, \ldots, n\}$ are assumed to follow a Pólya urn scheme. Using this representation the differences between any two given clusters can be summarised by a difference in only a subset of the attribute means, with the subset depending on the pair of clusters being compared.

2.5 Supervised Learning

So far we have discussed the role of mixture models as part of an unsupervised learning process and we focused our attention on recent studies performing cluster analysis with variable selection. Here we review the most significant literary contributions that show how the flexibility of mixture models can also be exploited in supervised learning.

The supervised learning process involves observing the training dataset $\mathcal{D}^n = (\boldsymbol{x}_i, y_i)_{i=1}^n$ where $y_i \in \mathbb{R}$ and $\boldsymbol{x}_i \in \mathbb{R}^p$, and proposing a rule that describes the dependency of set of response variables $\boldsymbol{y}$ on the corresponding input variables $\boldsymbol{x}$. In classification problems, the response variables are typically discrete class labels and the objective is to estimate the posterior probability of class membership of each observation so that it can be assigned to one of the finite number of categories. Regression problems are usually characterized by continuous real response variables. The objective of the regression modelling can be to explain the observed data, by finding the curve that best describes the relationship between input and response variables, or to make reasonable inference by predicting what should be the expected output when we are presented with previously unseen values of $\boldsymbol{x}$.

We first introduce the Gaussian and the logistic linear regression models as an example of viable methods to solve regression and classification problems respectively. We then discuss how these simple models can respond to the challenges of

more complex and high dimensional data by adopting a mixture structure. Since both the Gaussian and the logistic linear regression models belong to the comprehensive class of Generalized Linear Models (GLM) we start by reviewing the essential aspects of GLMs and introducing the necessary notation.

The Regression Model

In its most simple formulation, a supervised learning process aims to construct the regression function $m(\cdot)$ that links a random variable $Y \in \mathbb{R}$ to the explanatory variable $\boldsymbol{X} \in \mathbb{R}$ according to the model

$$Y = m(\boldsymbol{X}) + \epsilon$$

where ϵ is the additive stochastic component that usually reflects the fact that Y might also depend on other quantities that are neither controlled nor observed. We can interpret ϵ as an unobservable random error and assume that it follows a Gaussian distribution with $\mathbb{E}(\epsilon) = 0$ and $\text{Var}(\epsilon) = \sigma^2$ independently of the explanatory variable $\boldsymbol{X}$.

From a probabilistic perspective, the observed response is only one out of many possible outcomes that we could have observed under identical circumstances, and we describe the possible values in terms of a probability distribution. Considering $\mathcal{D} = (\boldsymbol{X}, Y)$ as being sampled from an unknown distribution $F_{\boldsymbol{X},Y}$, under the previous assumptions about ϵ, we can still assume that the conditional distribution of Y given $\boldsymbol{X}$ is a normal distribution:

$$Y|\boldsymbol{X} \sim \mathcal{N}(m(\boldsymbol{X}), \sigma^2).$$

The intuition then is that the regression function $m(\boldsymbol{x})$ can be defined as the conditional expectation $\mathbb{E}(Y|\boldsymbol{X} = \boldsymbol{x}) = \int y\, f_Y(y|\boldsymbol{x})\, dy$. In practice, the distribution of y does not have to be necessarily Gaussian, but as long as it is part of the exponential family, the regression model can be seen as a specific case of the Genaralized Linear Models.

2.5.1 Generalized Linear Model

Generalized linear models were introduced by Nelder and Wedderburn (1972) as a way of unifying various other statistical models, including linear regression, logistic regression and Poisson regression. In a GLM, each outcome of the dependent variables, Y, is assumed to be generated from a particular distribution in the exponential family which encompass a large range of probability distributions such as the normal, binomial, Poisson, gamma and inverse Gaussian, among others.

Each observation y_i for $i = \{1, \ldots, n\}$ that has been sampled from a distribution in the exponential family has the following probability density function

$$f(y_i) = \exp\left\{\frac{y_i\,\theta_i - b(\theta_i)}{a_i(\phi)} + c(y_i,\,\phi)\right\}$$

where θ_i and ϕ are parameters and $a_i(\phi)$, $b(\theta_i)$ and $c(y_i, \phi)$ are known functions that characterize each special distribution. In all models we consider here the function $a_i(\phi)$ has the form $a_i(\phi) = \phi/p_i$, where p_i is a known prior weight, usually 1. The parameters θ_i and ϕ are essentially location and scale parameters and it can be shown that if Y_i has a distribution in the exponential family then it has mean and variance

$$\mathbb{E}(Y_i) = \mu_i = b'(\theta_i)$$

$$\mathrm{Var}(Y_i) = \sigma_i = b''(\theta_i)\,a_i(\phi)$$

where $b'(\theta_i)$ and $b''(\theta_i)$ are the first and second derivatives of $b(\theta_i)$.

The link Function

In a GLM, it is assumed that

$$\begin{aligned}\eta_i &= g(\mu_i)\\ &= \boldsymbol{x}_i'\boldsymbol{\beta}\end{aligned}$$

where $\boldsymbol{x}_i$ is the vector of covariates or explanatory variables on the i^{th} response y_i and $\boldsymbol{\beta}$ is a vector of unknown regression coefficients. The one-to-one continuous differentiable transformation $g(\cdot)$ is a monotonic function known as the link function. The quantity η_i is called the linear predictor because it is expressed as a linear combination of the unknown parameters $\boldsymbol{\beta}$ and incorporates the information about the independent variables into the model.

Since the link function defines the relationship between the linear predictor and the mean of the distribution function, the systematic component of the regression model can be found as

$$\mu_i = g^{-1}(\boldsymbol{x}_i' \boldsymbol{\beta})$$

which does not have to be necessarily linear in the parameters since the linear predictor is transformed by the nonlinear function $g^{-1}(\cdot)$. This allows the model to have more complex analytical and computational properties than a simple linear regression models could have.

Furthermore, when setting the canonical parameter $\theta = \eta$, see Hastie and Tibshirani (1990), the canonical link function express θ in terms of μ and maps whatever specific density function we have chosen to its canonical GLM form. As we see next the identity is the canonical link function for the normal distribution while the logit is the canonical link function for the binomial distribution.

Normal Linear Regression

A simple, very important example of a generalized linear model is the normal linear regression, see for example (Nelder and Wedderburn, 1972; Hastie and Tibshirani, 1990; Figueiredo, 2000). The model treats the responses y_i as independent realizations of Gaussian random variables

$$Y_i \sim \mathcal{N}(\mu_i, \sigma_i). \tag{2.29}$$

Assuming the observed data $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$, where $y_i \in \mathbb{R}$ and $\boldsymbol{x}_i \in \mathbb{R}^p$, are i.i.d. samples, the Normal Linear model can be derived from the GLM by taking the

identity function as its canonical link function

$$\eta_i = \boldsymbol{x}_i' \boldsymbol{\beta} = \mu_i \tag{2.30}$$

which implies a mean function

$$\mu_i = \boldsymbol{x}_i' \boldsymbol{\beta}. \tag{2.31}$$

which shows how the mean μ_i depends linearly on the parameters $\boldsymbol{\beta}$ and on the covariates $\boldsymbol{x}$. The explicit formulation of the Normal linear regression model yields

$$\mathbb{E}(y_i|\boldsymbol{x}_i) = \beta_0 + \beta_1 x_{i,2} + \ldots + \beta_p x_{i,p} \tag{2.32}$$

where the first component of $\boldsymbol{x}_i$ vector is taken to be 1 and the corresponding first component of the coefficient vector $\boldsymbol{\beta}$ is relabelled β_0; in this case β_0 is known as the intercept term.

Logistic Regression Model

The logistic regression model is another specific case of GLM whose natural application is in classification problems, Hastie and Tibshirani (1990); McLachlan and Peel (2000); Bishop (2007). Logistic regression in fact is a common method for analysing the effect of a vector of covariates on the number of successes in a series of N_i independent Bernoulli trials.

Let us assume that $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$ y_i is a realization of a random variable $Y_i \in \{0, \ldots, N_i\}$. If the N_i observations in each group are independent, and they all have the same probability π_i of being a success, i.e. having the attribute of interest, then the distribution of Y_i is binomial with parameters π_i and N_i:

$$Y_i \sim \mathcal{B}(N_i, \pi_i).$$

The corresponding canonical parameter that links it to the exponential family is the logit of π_i

$$\theta_i = \log\left(\frac{\pi_i}{1-\pi_i}\right)$$

which yields the mean function $\mu_i = b'(\theta_i) = N_i \pi_i$. The logit function, which maps probabilities from the range $(0, 1)$ to the entire real line, is the canonical link function for a binomial distribution

$$\eta_i = \boldsymbol{x}_i' \boldsymbol{\beta} = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \log\left(\frac{\mu_i}{N_i - \mu_i}\right)$$

or equivalently

$$\pi_i = \frac{\exp(\boldsymbol{x}_i' \boldsymbol{\beta})}{1 + \exp(\boldsymbol{x}_i' \boldsymbol{\beta})}$$

From a practical point of view it is important to note that the Bernoulli distribution for the individual zero-one data or the binomial distribution for grouped data consisting of counts of successes in each group are equivalent approaches, in the sense that they lead to exactly the same likelihood function and therefore the same estimates and standard errors.

While parametric models such as linear regression remain the most popular techniques in data modelling, they are often too rigid to model general nonlinear patterns hidden in a high-dimensional data space. In response to that demand, a procedure for multivariate regression that is applicable to high dimensional data and retains the flexibility of nonparametric methods is reviewed in the next section.

2.5.2 Mixture of Regressions

There exist many areas where heterogeneous populations of covariates are found and studied (Goldfeld and Quandt, 1973; Quandt and Ramsey, 1978; Friedman, 1991; Figueiredo, 2000; Hurn et al., 2010). In similar circumstances, if the objective of the study is to classify observations, the focus of the learning process is on identifying the homogeneous subpopulations. If, instead, the objective is to explain the data and make prediction, the focus is rather on finding the most appropriate rule for each subpopulation. In both cases it is possible to extend the mixture model approach that we have seen implemented in unsupervised learning to the supervised learning process. Mixture of regressions provide a flexible tool to investigative the relationship between input and response variables coming from

several unknown latent components.

Mixture of Linear Regressions

Let us assume that for the observed dataset $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$ there are K possible regression curves and let $\boldsymbol{z}$ with $z_i \in \{1, \ldots, K\}$ be a latent class variable with $p(z_i = k|\boldsymbol{x_i}) = w_k$ for $k = 1, \ldots, K$. Given $z_i = k$ the observed quantity y_i depends on a vector of covariates $\boldsymbol{x_i}$ in a linear way according to the k^{th} rule:

$$y_i = \boldsymbol{x}_i' \boldsymbol{\beta}_k + \epsilon_{i,k}$$

where the parameters $\boldsymbol{\Theta}_k = (\boldsymbol{\beta}_k, \sigma_k)$ vary among a set of K possible values with probabilities $w_1, \ldots, w_K$. In other words, assuming a normal distribution on the perturbation, it is possible to model the joint density of y_i and $\boldsymbol{x}$ as Gaussian mixture which also allows to express the conditional distribution of y_i given $\boldsymbol{x}$ as a mixture of normal distributions

$$p(y_i|\boldsymbol{x}_i) = \sum_{k=1}^{K} w_k \mathcal{N}(\boldsymbol{x}' \boldsymbol{\beta}_k, \sigma_k^2).$$

The Gaussian Mixture of regressions in this case has the tight and parsimonious structure of a parametric model, yet it still retains the flexibility of a nonparametric method. In figure 2.5.2 we see an example of a mixture of three $1-$dimensional regression curves.

Mixture of Logistic Regressions

Following the same approach, we can solve the problem of classifying an heterogeneous population by using the simple logistic regression model as a component of a mixture of logistic regressions, see Bishop (2007); Hurn et al. (2010).

An obvious extension of the binomial conditional probability for the response

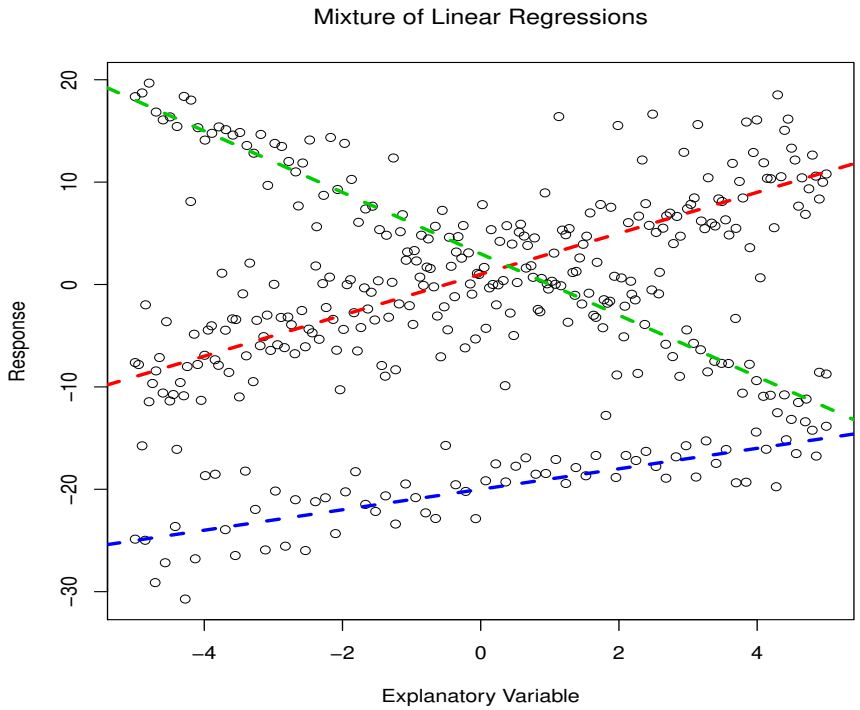

Figure 2.1: Mixture of three linear regression curves with different mixing proportions

variable y_i yields

$$p(y_i = 1|\boldsymbol{x}) = \sum_{k=1}^{K} w_k \left[\frac{\exp(\boldsymbol{x}'\,\boldsymbol{\beta}_k)}{1 + \exp(\boldsymbol{x}'\,\boldsymbol{\beta}_k)} \right]$$

where the parameter $\pi_{i,k}$ that describe the probability of success within each component is postulated to depend on the explanatory variables according to the equation

$$\log\left(\frac{\pi_{i,k}}{1-\pi_{i,k}}\right) = \boldsymbol{x}_i'\,\boldsymbol{\beta}_k.$$

Estimation Procedures

It is straightforward to extend the maximum likelihood or the Bayesian approach we have discussed for unsupervised mixture models to the estimation process of the mixtures of regression and logistic models' parameters.

The standard method of obtaining maximum likelihood (ML) estimates is still

the expectation maximization (EM) algorithm, see Wedel and DeSarbo (1995).

In the Bayesian approach independent normal priors are usually proposed for the regression coefficients $\boldsymbol{\beta}_k$, see Hurn et al. (2010). The choice of a standard conjugate prior for $\boldsymbol{\beta}_k$ together with inverse gamma prior on the scale parameter σ_k and a Dirichlet prior on the mixing proportion w_k would allow us to use a Gibbs sampler, but other choices are equally acceptable since they can be implemented via alternative MCMC methods.

The ordinary Least Squares Estimator (LSE), instead, is peculiar to the supervised learning framework. The LSE estimates of the regression coefficients, $\hat{\boldsymbol{\beta}}$, are found by minimizing the square difference between the observed response y_i and the response $\hat{y}_i$ suggested by the regression rule. While the LSE has minimum variance among unbiased estimators and is an efficient estimator if the error is normal it is also extremely sensitive to outliers and the heavy tailed error distribution.

2.6 Variable Selection in Supervised Learning

The practical utility of variable selection is well recognised in regression and classification models, see for older and more recent example (Tibshirani, 1996; George and Mcculloch, 1997; Tipping, 2001; Khalili and Chen, 2007; Hans, 2009; Fahrmeir et al., 2009; Lee et al., 2010; Schäfer and Chopin, 2011). Often, there are many covariates of interest whose contributions to the response variable might be small in same instances or null in others. To enhance the parsimony of the model without compromising its accuracy, it is common practice to include only the important covariates in the model.

In a mixture models framework, the underlying idea is that a single set of regression coefficients across all observations may be inadequate and potentially misleading if the observations arise from a number of unknown groups in which the coefficients differ (Wedel and DeSarbo, 1995).

Most of the variable selection methods discussed in supervised learning literature stem from the same intuitions behind the variable selection methods we have presented in section 2.4. Here we review some significant examples applied to mix-

ture of regression models, spanning Penalised Maximum Likelihood and Bayesian sparsity-inducing priors approaches.

To formalize the problem in general terms, let us assume that the recorded dataset $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$ with $y_i \in \mathbb{R}$ and $\boldsymbol{x}_i \in \mathbb{R}^p$ has been sampled from an heterogeneous population. It is fair to assume that there are K possible regression curves which describe the data and that each curve depends upon a different collection of the variables $1, \ldots, p$. We will write $\boldsymbol{\gamma}_k$ as the p-dimensional vector of zeros and ones which denote whether or not variable $d \in \{1, \ldots, p\}$ is retained by the k^{th} regression curve.

When the explanatory variable x_d is not significant, the corresponding estimated regression coefficient β_d is often close to, but not equal to 0. Thus this covariate would typically not be excluded from the model. To avoid this problem, we may explore several submodels based on different subsets of variables and compare them according to an information criteria such as the Akaike criterion, (Akaike, 1973), or the Bayes criterion, (Schwarz, 1978). However, the computational burden of these approaches is computationally intensive and very expensive as the number of covariates and components in the mixture model increases.

Penalised Maximum Likelihood Approach

Unlike the subset selection methods, a penalised maximum likelihood approach can be used to tackle variable selection problems in reasonably high-dimensional datasets.

The original idea was proposed and applied to a simple normal linear regression by Tibshirani (1996). In his work variable selection was performed by choosing only the largest coefficients in absolute value and setting the rest to 0. For some choice of $\lambda \in \mathbb{R}^+$ the regression coefficient β_d was forced to collapses to 0 if $|\hat{\beta}_d| < \lambda$ where the estimate $\hat{\boldsymbol{\beta}}$ are found minimizing:

$$\sum_{i=1}^{n} \left(y_i - \boldsymbol{x}_i' \boldsymbol{\beta}\right)^2 + \lambda \boldsymbol{\beta}^2$$

The intuition of adding to the likelihood function a penalty term that depends on the size of the regression coefficients, $\log L_p(\boldsymbol{\Psi}) = \log L(\boldsymbol{\Psi}) - h_{\boldsymbol{\lambda}}(\boldsymbol{\beta})$, can naturally be extended to a mixture of regressions, see Khalili and Chen (2007). The penalisation can also be made dependent on class membership defining the penalty function

$$h_{\boldsymbol{\lambda}}(\boldsymbol{\beta}) = \sum_{k=1}^{K} w_k \left\{ \sum_{d=1}^{p} h_{\lambda_k}(\beta_{k,d}) \right\}$$

where the $h_{\lambda_k}(\beta_{k,d})$ values are nonnegative and nondecreasing functions in $|\beta_{k,d}|$. By maximizing the penalised log-likelihood $\log L_p(\boldsymbol{\Psi})$ there is a positive chance of having some estimated values of $\boldsymbol{\beta}$ equalling 0 and thus of automatically selecting a submodel.

Different form of penalty functions have been presented and discussed in the literature. Besides the LASSO which is convex and thus advantageous for numerical computation, other functions have been considered in order to reduce bias on larger coefficients and compromising between sparsity and continuity, see for examples Khalili and Chen (2007); Fan and Li (2001); Law et al. (2004). The properties that characterize each alternative form of $h_{\lambda_k}(\cdot)$ are different degrees of unbiasedness, sparsity, continuity.

In all cases the penalty functions are designed to be dependent on the size of the regression coefficients and the mixture structure and also share the noticeable advantage that can fit very well with the EM framework.

Bayesian Approach

The key feature of the Bayesian approach is that as well as offering good generalisation performance, the inferred predictors are exceedingly sparse in that they contain relatively few non-zero coefficients parameters $\boldsymbol{\beta}$. The majority of parameters are automatically set to zero during the learning process, giving a procedure that is extremely effective at discerning those variables which are relevant for making good predictions.

In a fully probabilistic framework it is possible to introduce a prior over the

model weights governed by a set of hyperparameters. Sparsity is achieved because for higher dimensional dataset we usually find that the posterior distributions of many of the weights are heavily peaked around zero.

A popular choice, see Tipping (2001), is to define a zero-mean Gaussian prior distribution over $\boldsymbol{\beta}$:

$$p(\boldsymbol{\beta}|\boldsymbol{\alpha}) = \prod_{d=0}^{p} \mathcal{N}(\beta_d|0, \alpha_d^{-1}) \tag{2.33}$$

where $\boldsymbol{\alpha}$ is a $p+1$ dimensional vector of hyperparameters each one independently controlling the strength of the prior on the associated input variable. To fully specify the hierarchical structure, a Gamma prior is proposed for the hyperparameter:

$$p(\boldsymbol{\alpha}) = \prod_{d=0}^{p} \mathcal{G}a(\alpha_d^{-1}|a, b). \tag{2.34}$$

This approach allows us to produce a smoother and at the same time less complex regression function.

Explicit Bayesian variable selection is commonly based on spike and slab priors for regression coefficients (George and Mcculloch, 1997; Tadesse et al., 2005; Kim et al., 2006; Schäfer and Chopin, 2011). For variable selection in this case we specifically refer to inclusion or exclusion of covariates x_d with linear effects $x_d\,\beta_d^k$ as part of the linear predictor $\boldsymbol{x'}\,\boldsymbol{\beta}^k$.

These priors take the form of a finite mixture distribution with two components where one component (the spike) is centered at 0 and shows little variance compared to the second component (the slab) which has considerably larger variance. Spike-and-slab priors can easily be extended to variable selection for random intercept model and lead a two component mixture prior for β_d :

$$p(\beta_d|\gamma_d, \boldsymbol{\theta}) = (1-\gamma_d)p_{spike}(\beta_d|\boldsymbol{\theta}) + \gamma_d\, p_{slab}(\beta_d|\boldsymbol{\theta}) \tag{2.35}$$

We assume that β_d are independent a priori conditional on the hyperparameters γ_d and $\boldsymbol{\theta}$.

Chapter 3

Penalised Mixtures of Student's t Distributions

3.1 Introduction

In chapter 1 we have described the real life problems we intend to investigate and declared our research objective, that is to make inference about the cluster membership of each observation and identify the subset of relevant features. As we reviewed the literature on unsupervised learning methods delivering a sparse clustering solution, in chapter 2, we found that mixture models are particularly well suited to model high dimensional heterogeneous data and that penalised likelihood is a valid variable selection procedure for this class of models.

Following on this promising approach, in this chapter we propose a penalised mixture of Student's t distributions model that simultaneously (a) identifies the informative variable and ranks them by importance, and (b) discovers clusters that may exist in the datasets when considering the selected features only. As observed before, the use of group-specific distributions having longer tails results in robust clustering assignment that are less prone to be affected by extreme or unusual observations, and facilitates the discovery of the true underlying number of clusters. Variable selection is achieved by imposing an adaptive $\mathbb{L}_1$-norm penalty function acting on the location and the dispersion parameter. Moreover, the data

resampling procedure we introduce, allows us to quantify the contribution of each feature to the clustering process, thus providing a natural metric for ranking the features, and improves on the selection of the true number of clusters.

The chapter is organised as follows. In section 3.2 we discuss the benefits of robust modelling and introduce the proposed penalised finite mixture of Student's t distributions. In section 3.3 we exploit the hierarchical representation of the mixture models and derive a specific EM algorithm to estimate the unknown parameters of the model. In section 3.4 we illustrate the model selection problem and present a resampling procedure that, when combined with the standard Bayesian information criteria, enhances the model selection accuracy. In section 3.5 using experimental data, we asses how well the proposed methodology performs under different demanding scenarios. We analyse its clustering and variable selection accuracy in comparison with two competing algorithms and illustrate in which situations our model is expected to perform better. We also verify that the resampling procedure can effectively improve model selection and discuss how it can offer an insight on the relative importance of each selected variable.

3.2 The Model

As we noted in chapter 2, despite the popularity of mixture models based on Gaussian components, this particular choice of probability distributions may not always be ideal.

Robust Modelling

The assumption that the observed data have a normal distribution, has played a critical part in statistical literature since the development of this discipline and has been the preferred framework for most of the classical methods in supervised and unsupervised learning. The central limit theorem is a valid theoretical argument for assuming a normal distribution, but it is also quite convenient in practice that Gaussian density function allows explicit tractable formulae to be derived in several optimization problems such as maximum likelihood.

Often probabilistic models based on the Gaussian distribution are able to accurately describe the majority of the observations, but perform poorly when some observations follow a different pattern or no pattern at all. The reality is that, while the behaviour of many datasets appear to be rather normal, this is held only approximately. There might still be a small proportion of observations which are outliers, such that while the observed distribution has a normal shape in the central region, the tails are fatter than we expect a normal distribution to have. The consequence is that, when the data are assumed to be normally distributed but their actual distribution has heavy tails, then estimates based on the maximum likelihood principle might not be the best estimate and have unnecessarily large variance, if the tails are symmetric, or may have very large bias, if the tails are asymmetric.

In the presence of high measurement noise, a mixture model that assumes Gaussian components may suffer from a lack of robustness against outliers, as we experienced with microarray data. In similar situations, the model might be misled to fit spurious clusters in order to capture the heavy tail distributions that characterise certain groups and thus suggest an inflated number of detected clusters (Qu and Xu, 2004; He et al., 2006; Liu and Rattray, 2010).

The robust approach to statistical modelling and data analysis aims at deriving methods that produce reliable parameter estimates, not only when the data follow a given distribution exactly, but also when this happens only approximately in the sense just described above. More precisely, if the data contain no outliers, the robust method gives approximately the same results as the classical method. On the other hand, if a small proportion of outliers are present, the robust method returns the same results as the classical method applied on the bulk of the data with the outliers removed.

Mixture of Multivariate Student's t distributions

As an alternative to normally distributed components, Student t distributions have been successfully used for robust model-based clustering in several application

domains, including the analysis of gene expression data (Liu and Rubin, 1995; Peel and McLachlan, 2000; Jiao and Zhang, 2008; Jiao, 2010).

The Student's t density function can achieve slower exponential tail decay, thus yielding heavier tails which makes it more robust to outliers or sampling errors. By reducing the weight assigned to extreme observations, we are able to estimate more accurately the location and dispersion parameters and improve the clustering performance of the mixture model, see Peel and McLachlan (2000) for mixtures of t and Lin et al. (2007) for mixture of skew t implementation.

In its canonical form, the density function of a p-dimensional random variable following a Student's t distribution is:

$$\mathcal{S}t(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) = \frac{\Gamma((\nu + p)/2)|\boldsymbol{\Sigma}|^{-1/2}}{\Gamma(\nu/2)(\pi\nu)^{p/2}\{1 + \delta(\boldsymbol{y}|\,\boldsymbol{\mu},\, \boldsymbol{\Sigma})/\nu\}^{(\nu+p)/2}}, \tag{3.1}$$

where $\Gamma(\cdot)$ is the gamma function and $\delta(\boldsymbol{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = (\boldsymbol{y} - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{\mu})$ is the Mahalanobis squared distance. The set of unknown parameters $\boldsymbol{\Theta}$ includes the p-dimensional location vector $\boldsymbol{\mu}$, a $p \times p$ covariance matrix $\boldsymbol{\Sigma}$ and the degrees of freedom parameter ν. In the following discussion we will assume a diagonal covariance matrix, $\boldsymbol{\Sigma} = diag(\sigma_1^2, \ldots, \sigma_p^2)$ with $\boldsymbol{\sigma} = \{\sigma_1, \ldots, \sigma_p\}$ the p-dimensional vector of standard deviations. This assumption simplifies the computations and makes the proposed algorithm scale to very high dimensional settings without hindering its ability to select the truly informative variables, (Xie et al., 2008a, for instance).

For a more comprehensive discussion and less canonical representation of the multivariate Student's t distribution we refer the reader to the book of Kotz and Nadarajah (2004). Here, we explicitly derive only the log-likelihood function, which will be extensively used in the following discussion. Given a collection of p-dimensional t distributed samples $\boldsymbol{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$, the log-likelihood function

is

$$\begin{aligned}\log\, L(\boldsymbol{\Theta}) &= \sum_{i=1}^{n} \log \mathcal{S}t(\boldsymbol{y}_i|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) \\ &= -\frac{1}{2}\, n\, p \log(\pi\nu) + n\left[\log\Gamma(\nu + \frac{p}{2}) - \log\Gamma(\frac{\nu}{2})\right] - \frac{1}{2}\, n\, p \log|\boldsymbol{\Sigma}| \\ &\quad -\frac{1}{2}\, n\, (\nu + p) \log(\nu) - \frac{1}{2}\, (\nu + p) \sum_{i=1}^{n} \log\, [\nu + \delta(\boldsymbol{y}_i|\boldsymbol{\mu}, \boldsymbol{\Sigma})]\end{aligned}$$

Using the standard mixture model notation, where f_k is the Student's t density function of the k-th component and $\boldsymbol{\Theta}_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k\}$ the associated density parameters, the robust model we propose yields

$$p(\boldsymbol{y}) = \sum_{k=1}^{K} w_k\, f_k(\boldsymbol{y}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k) \tag{3.2}$$

and the corresponding log-likelihood is

$$\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \left[\log\left(\sum_{k=1}^{K} w_k\, f_k(\boldsymbol{y}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k)\right)\right] \tag{3.3}$$

where the set of all unknown parameters $\boldsymbol{\Psi}$ includes $\boldsymbol{\Theta} = (\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_K)$ and $\boldsymbol{w} = (w_1, \ldots, w_K)$.

3.2.1 Variable Selection through Penalized Log Likelihood

Other than the presence of measurement errors and outliers, we mentioned in chapter 2 that another potential obstacle to standard clustering algorithms is posed by the fact that not all the recorded features will necessarily contribute equally to the identification of distinct sub-populations. Even when real clusters exist and are well separated, it is often the case that only a subset of variables will show a significantly different distribution across groups. Failing to identify the truly informative variables may yield inaccurate clustering results since the non-informative variables will mask the underlying structure of the data.

In order to reduce the misleading effect that noisy non-informative variables

might have on the clustering performance of our model, we resolve to implement a variable selection approach that suits the mixture of distributions framework. Even if the penalised likelihood methodology has originally been proposed to regularize coefficients in regression problems, this approach has recently been implemented with success in several unsupervised mixture models, as noted in section 2.4.1. Nonetheless, most of the literature so far has considered only the standard case of Gaussian components. We then propose to extend the penalised likelihood approach to robust models and perform variable selection in the context of mixtures of Student's t distributions.

Before describing the solution in detail, let us review the general settings, using the previously introduced notation. In chapter 2, we assumed that only $1 \leq m \leq p$ variables have been generated by distinct density functions and therefore suggested that only those variables are informative and should contribute to the clustering process. The remaining $q = p - m$ variables which are considered non informative, should then be excluded from the model.

Under such circumstances, we can effectively perform variable selection by imposing some form of regularization on the most relevant parameters that characterize the density functions of each one of the K components. In practice, a sparse clustering solution is achieved by adding a penalty term to the log-likelihood function (3.3) before it is maximized to find the best estimates of $\boldsymbol{\Psi}$:

$$\log L(\boldsymbol{\Psi}) = \sum_{i=1}^{n} \left[\log \left(\sum_{k=1}^{K} w_k \, f_k(\boldsymbol{y}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k) \right) \right] - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_K) \qquad (3.4)$$

where $h(\cdot)$ is a penalty function that depends on the parameters of the mixture components as well as on a regularisation parameter vector $\boldsymbol{\lambda}$.

Once we have adopted a penalised maximum likelihood approach, choosing a suitable penalty function becomes a critical decision. We have seen in section 2.4.1 that several different options have been suggested by different studies. In this thesis, we propose a joint adaptive penalty function that combines the contribution of different studies (Pan and Shen, 2007; Xie et al., 2008a), and enforces simultaneous

regularization of the location and dispersion parameter

$$h_{\boldsymbol{\lambda}} = \lambda_{\mu} \sum_{k=1}^{K} \sum_{d=1}^{p} \omega_{k,d}^{(\mu)} \, |\mu_{k,d}| + \lambda_{\sigma} \sum_{k=1}^{K} \sum_{d=1}^{p} \omega_{k,d}^{(\sigma)} \, |\log \sigma_{k,d}^{2}| \tag{3.5}$$

where $\boldsymbol{\lambda} = (\lambda_{\mu}, \lambda_{\sigma})$. In other words, the log-likelihood of the model is penalised proportionally to the absolute values of $\boldsymbol{\mu}$ and $\log \boldsymbol{\sigma}^2$, where $\lambda_{\mu} \in \mathbb{R}^{+}$ and $\lambda_{\sigma} \in \mathbb{R}^{+}$ are the regularisation parameters for mean vector and covariance matrix, respectively.

The intuition behind (3.5) is that the non-informative variables can be described by one general distribution which is valid for all the samples irrespective of their cluster membership. Under this assumption and given that the data have been previously standardized, for every $d \in \{1, \ldots, p\}$, any deviation of the estimated values of $\mu_{k,d}$ from 0 and $\sigma_{k,d}^2$ from 1 in one particular cluster will be interpreted as noise. Only the informative variables justify increasing the complexity of the model by demanding distinct parameters in different clusters. Therefore, the likelihood of the model will only depend on the cluster assignment of those variables whose location and dispersion parameter resist the shrinkage across all components and do not collapse to 0 and 1 respectively.

On the other hand, the remaining variables, whose parameters have collapsed towards the population average, will yield the same likelihood in every cluster. Since the contribution of the non-informative variables to the likelihood of the model is independent of the proposed clustering, they will not be able to influence the assignment of the observations to any particular group.

Note that, since ML estimation with the L_1-norm penalty function has been proved to produce biased estimates of large parameters (Zou, 2006), to attack this problem we implement an adaptive versions of penalty function. While λ_{μ} and λ_{σ}, which control the module of the shrinkage, are set generally, adaptive weights are used for penalising different coefficients. The strategy consists in applying some penalisation weights such that the parameters of the densities corresponding to the informative variables receive less shrinkage. The rule we opt to follow sets each

weight $\omega_{k,d}^{(\mu)} = 1/|\mu_{k,d}|$ and $\omega_{k,d}^{(\sigma)} = 1/|\log \sigma_{k,d}|$ respectively.

As a final remark, it should also be pointed out that the penalties imposed on $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in (3.5) are additive and independent. While they could be implemented separately, we will let the model selection procedure indicate whether one or both jointly are necessary to identify non informative variables.

3.2.2 Hierarchical Representation

Having defined a suitable penalty function (3.5), we can estimate the unknown parameters of the model $\boldsymbol{\Psi}$ by maximizing the penalised log-likelihood function (3.4). While the optimization problem is computationally difficult because it requires to maximize the log of the sum over all the density functions, a more convenient formulation is possible in a missing data framework, as we have seen in section 2.3.2.

A K-dimensional component-label vector $\boldsymbol{z}_i = (z_{i,1}, \ldots, z_{i,K})$ is introduced to indicate the cluster membership of y_i for $i = (1, \ldots, n)$. With probability w_k, the value of $z_{i,k}$ is one if the observation y_i belongs to component k and zero otherwise. Then $\boldsymbol{z}_i$ follows a multinomial distribution with parameters $(w_1, \ldots, w_K)$. In (2.9) we have shown that by introducing latent variables $\boldsymbol{z}$ we are able to derive a hierarchical representation of the mixture model where the log-likelihood function factorises into a sum of logs, and can be more easily maximised.

Since we know that, conditional on $z_{i,k} = 1$, the marginal distribution of y_i is f_k, we can exploit the fact that the Student's t distribution itself admits a hierarchical representation and further simplify the log likelihood function of the model. Note, in fact, that the same distribution law of f_k can be obtained integrating an infinite number of Gaussian distribution whose precision scales proportionally to a gamma distributed variable u_i (Bishop, 2007). Considering the precision factor $\boldsymbol{u} = (u_1, \ldots, u_n)$ as another latent variable, the complete data set becomes

$\boldsymbol{y}_c = \{\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{u}\}$ and the hierarchical structure of the mixture model

$$\begin{aligned} \boldsymbol{z}_i &\sim \mathcal{M}(1|w_1, w_2, \ldots, w_K) \\ u_i|z_{i,k} = 1 &\sim \mathcal{G}a\left(\frac{\nu_k}{2}, \frac{\nu_k}{2}\right) \\ \boldsymbol{y}_i|u_i, z_{i,k} = 1 &\sim \mathcal{N}\left(\boldsymbol{\mu}_k, \frac{\boldsymbol{\Sigma}_k}{u_i}\right) \end{aligned}$$

leads to a more tractable penalised log-likelihood function which can be factorized as

$$\log L_p(\boldsymbol{\Psi}) = l_1(\boldsymbol{w}) + l_2(\boldsymbol{\nu}) + l_3(\boldsymbol{\mu}, \boldsymbol{\Sigma}) - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta}) \tag{3.6}$$

where

$$\begin{aligned} l_1(\boldsymbol{w}) &= \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k} \log w_i \\ l_2(\boldsymbol{\nu}) &= \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k}\{-\log\Gamma(\frac{\nu_k}{2}) + \frac{1}{2}\log\Gamma(\frac{\nu_k}{2}) \\ &\quad + \frac{\nu_k}{2}(\log u_i - u_i) - \log u_i\} \\ l_3(\boldsymbol{\mu}, \boldsymbol{\Sigma}) &= \sum_{i=1}^{n}\sum_{k=1}^{K} z_{i,k}\{-\frac{p}{2}\log(2\pi) + \frac{p}{2}\log(u_i) - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| \\ &\quad - \frac{1}{2}\, u_i\, (\boldsymbol{y}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu}_k)\} \end{aligned}$$

and $h_\lambda(\boldsymbol{\Theta})$ is as given in (3.5).

We should remark that in (3.6) we have been able to break down the log-likelihood of the complete data set $\boldsymbol{y}_c$ into a sum of different terms, each one depending on different parameters. More precisely $l_1(\boldsymbol{w})$ corresponds to the multinomial distribution of the latent variable $\boldsymbol{z}$ which depends on the mixing weights $\boldsymbol{w}$, the second term $l_2(\boldsymbol{\nu})$ is the log of the gamma distribution of the scaling factor $\boldsymbol{u}$ expressed as a function of the degrees of freedom $\boldsymbol{\nu}$ and finally $l_3(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the log of the distribution of the sampled data, which is Gaussian for a given value of the two latent variables.

Taking advantage of this convenient hierarchical representation of a mixture of

Student's t distributions we can derive a specific EM algorithm that can efficiently find the MLE of the model.

3.3 The EM Algorithm

We have suggested that a mixture of Student's t distributions (3.2) is a more suitable model to robustly perform cluster analysis on real life data. We later proposed a penalty function (3.4) to enforce shrinkage on the relevant parameters in order to perform variable selection. To then find the best estimate of the unknown parameters $\boldsymbol{\Psi} = (\boldsymbol{w}, \boldsymbol{\Theta})$, we derived a more tractable hierarchical representation of the model in (3.6) that we now need to maximize. As commonly done in similar settings, we propose an expectation-maximization algorithm (Dempster et al., 1977; McLachlan and Krishnan, 2008). Whilst a closed form solution for mixture models would be otherwise prohibitive given such high dimensional problem, the EM algorithm provides a computationally efficient methods for finding the MLE of the unknown parameters.

The aim of the EM algorithm is to explore the entire parameters space to find the values that maximizes the complete data density function (3.6). Starting from a randomly proposed value $\boldsymbol{\Psi}^0$, at each iteration t, the algorithm first evaluates the expectation of the penalised log-likelihood function of the complete data conditional on $\boldsymbol{\Psi}^{(t-1)}$ and then returns the updated MLE estimate $\boldsymbol{\Psi}^t$. By alternating the expectation and maximization steps, Dempster et al. (1977) demonstrated that the algorithm is bound to increase the log-likelihood of the model $\log L_p(\boldsymbol{y}_c|\boldsymbol{\Psi}^t) \geq \log L_p(\boldsymbol{y}_c|\boldsymbol{\Psi}^{t-1})$, until convergence.

As a preliminary remark, note that the penalisation we impose on the likelihood function does not alter the standard expectation step, since no latent variables is penalised directly. The maximisation step, instead, has to be modified to accommodate for the constraints imposed by the penalty function on the location and dispersion parameters.

The conditional expectation of the complete data taken with respect to the

posterior probability of the latent variables is

$$Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t-1)}) = E_{\boldsymbol{\Psi}^{(t-1)}}\{\log L_c(\boldsymbol{\Psi})|\, \boldsymbol{y}\} - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta}) \tag{3.7}$$

where $\mathrm{E}_{\boldsymbol{\Psi}^{(k-1)}}\{\log L_c(\Psi)|\, \boldsymbol{y}\}$ is the conditional expectation of the log-likelihood. It can be noted that the penalisation term $h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta})$ does not depend on any latent variable and it is a constant under the expectation. Using the hierarchical representation of the mixture of Student's t distributions in (3.6), the conditional expectation of the log-likelihood can be decomposed as

$$\begin{aligned} Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t-1)}) &= E_{\boldsymbol{\Psi}^{(t-1)}}\{\log l_{1,c}(\boldsymbol{w})\} + E_{\boldsymbol{\Psi}^{(t)}}\{\log l_{2,c}(\boldsymbol{\nu})\} + E_{\boldsymbol{\Psi}^{(t)}}\{\log l_{3,c}(\boldsymbol{\mu}, \boldsymbol{\Sigma})\} - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta}) \\ &= Q_1(\boldsymbol{w}|\boldsymbol{\Psi}^{(t-1)}) + Q_2(\boldsymbol{\nu}|\boldsymbol{\Psi}^{(t-1)}) + Q_3(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\Psi}^{(t-1)}) - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta}) \end{aligned} \tag{3.8}$$

Since all terms are additive and depend on different parameters, we can maximize each contribution independently in order to maximize Q_p.

3.3.1 The E Step

In the E-step, at the t^{th} iteration, the expected values of the latent variables are derived from their conditional posterior distribution given $\boldsymbol{\Psi}^{(t-1)}$.

The first conditional expectation term, $Q_1(\boldsymbol{w}|\boldsymbol{\Psi}^{(t-1)})$, corresponds to sum of the log of the multinomial distribution over all values of $\boldsymbol{z}$ weighted by the posterior probability of $\boldsymbol{z}$ conditional on $\boldsymbol{y}$ and $\boldsymbol{\Psi}^{(k)}$.

$$Q_1(\boldsymbol{w}|\boldsymbol{\Psi}^{(t-1)}) = E_{\boldsymbol{\Psi}^{(t-1)}}\left\{\sum_i^n \sum_k^K z_{i,k} \log w_k\right\} = \sum_i^n \sum_k^K E_{\boldsymbol{\Psi}^{(t-1)}}(z_{i,k}) \log w_k$$

where the conditional expectation of the indicator variable $\boldsymbol{z}_{i,k}$ is:

$$\begin{aligned} E_{\boldsymbol{\Psi}^{(t-1)}}\left(z_{i,k}|\, \boldsymbol{y}_i\right) &= z_{i,k}\, f_k(z_{i,k}|\boldsymbol{y}_i, \boldsymbol{\Psi}^{(t-1)}) \\ &= \frac{w_k^{(t-1)}\, f_k(\boldsymbol{y}_i|\boldsymbol{\mu}_k^{(t-1)}, \boldsymbol{\Sigma}_k^{(t-1)}, \nu_k^{(t-1)})}{f(\boldsymbol{y}_i|\boldsymbol{\Psi}^{(t-1)})} \equiv \tau_{i,k}^{(t)}. \end{aligned} \tag{3.9}$$

Similarly, the expected value of the other latent variables we introduced in (3.7) can be computed explicitly from their conditional posterior distribution. Since the conditional distribution of the scaling factor u_i is the gamma distribution, its conditional expected value is:

$$E_{\boldsymbol{\Psi}^{(t-1)}}\left(u_i|\,\boldsymbol{y}_i, z_{i,k}=1\right) = \frac{\nu_k^{(t-1)}+p}{\nu_k^{(t-1)}+\delta(\boldsymbol{y}_i|\,\boldsymbol{\mu}_k^{(t)},\,\boldsymbol{\Sigma}_k^{(t)})} \equiv u_{i,k}^{(t)}. \tag{3.10}$$

Finally, the conditional expected value of the log precision factor, $\log u_i$, is

$$E_{\boldsymbol{\Psi}^{(t-1)}}\left(\log u_i|\,\boldsymbol{y}_i, z_{i,k}=1\right) = \log u_{i,k}^{(t)} + \left\{\psi\left(\frac{\nu_k^{(t-1)}+p}{2}\right) - \log\left(\frac{\nu_k^{(t-1)}+p}{2}\right)\right\} \tag{3.11}$$

where ψ is the digamma distribution $\psi(s) = \{\partial\Gamma(s)/\partial s\}/\Gamma(s)$.

The latent variables can now be replaced in (3.8) with their expected values (3.9), (3.10), (3.11) and obtain

$$\begin{aligned}
Q_1(\boldsymbol{w}|\boldsymbol{\Psi}^{(t-1)}) &= \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{i,k}^{(t)}\,\log w_k \\
Q_2(\boldsymbol{\nu}|\boldsymbol{\Psi}^{(k)}) &= \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{i,k}^{(t)}\,Q_{2,k}(\nu_k|\boldsymbol{\Psi}^{(t-1)}) \\
Q_3(\boldsymbol{\mu},\Sigma|\boldsymbol{\Psi}^{(t-1)}) &= \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{i,k}^{(t)}\,Q_{3,k}(\boldsymbol{\mu}_k\,\boldsymbol{\Sigma}_k|\boldsymbol{\Psi}^{(t-1)})
\end{aligned}$$

where the complete expression for $Q_{2,k}(\boldsymbol{\nu}_k|\boldsymbol{\Psi}^{(t-1)})$ and $Q_{3,k}(\boldsymbol{\mu}_k,\,\boldsymbol{\Sigma}_k|\boldsymbol{\Psi}^{(t-1)})$ are given in the Appendix section 3.A.1.

3.3.2 The M Step

In the M-step the updated estimate of $\boldsymbol{\Psi}$ is found by maximising (3.7) given $\boldsymbol{\Psi}^{(t-1)}$ and given the expected values of the latent variables computed in the E-step, that is

$$\boldsymbol{\Psi}^{(t)} = \arg\max_{\boldsymbol{\psi}} Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t-1)}) \tag{3.12}$$

If the solution $\boldsymbol{\Psi}^t$ exists, then the following inequality holds true:

$$Q_p(\boldsymbol{\Psi}^{(t)}|\boldsymbol{\Psi}^{(t-1)}) \geq Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t-1)})$$

which also implies $\log L_p(\boldsymbol{\Psi}^t) \geq \log L_p(\boldsymbol{\Psi}^{t-1})$, as shown by Dempster et al. (1977). This conclusion guarantees that iterating the EM algorithm the updated estimates of $\boldsymbol{\Psi}$ will eventually converge to the penalised MLE.

Since all terms in (3.8) are additive and depend on different parameters, we can solve (3.12) by maximising each term separately. The new estimated value of $\boldsymbol{w}$ is the root of the derivative of $Q_1(\boldsymbol{w}|\boldsymbol{\Psi}^{(t-1)})$ with respect to $\boldsymbol{w}$

$$\frac{\partial}{\partial \boldsymbol{w}} \left\{ \sum_{i=1}^{n} \sum_{k=1}^{K} \tau_{i,k}^{(t)} \log w_k \right\} = 0.$$

Using a Lagrange multiplier to enforce the constraint $\sum_{k=1}^{K} w_k = 1$, the update for w_i is

$$w_k^{(t)} = \sum_{i=1}^{n} \tau_{i,k}^{(t)} / n.$$

The estimated mixing weights are then proportional to the *responsibility* of each component, that is the relative frequency of observations assigned to that component.

The term $Q_2(\boldsymbol{\nu}|\boldsymbol{\Psi}^{(t-1)})$ is a function of the degrees of freedom. No closed form solution is available for the first derivative with respect to ν_k

$$\frac{\partial}{\partial \nu_k} \left\{ \sum_{i=1}^{n} \tau_{i,k}^{(t)} \, Q_{2j}(\nu_k|\boldsymbol{\Psi}^{(t-1)}) \right\} = 0$$

Nonetheless, as we can see from its explicit formulation in appendix 3.A.2, the first derivative is still a function smooth enough to have a straightforward numerical solution that can be found by any standard optimisation algorithm. In our implementation, we use the PORT routines in the R package `nlminb`.

Updating Step for $\boldsymbol{\mu}$

The third term $Q_3(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\Psi}^{(k-1)})$ is the only one depending on parameters that are subject to regularisation. In this case we set up a constrained maximisation problem that takes into consideration the relevant penalty term. Here we report the essential steps that lead to the updating algorithms for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$, we refer the interested reader to section 3.A.3 and 3.A.4 in the appendix for a full account and proof of how the algorithms are derived.

First, an update for the penalised mean vector $\boldsymbol{\mu}$ is obtained by finding the maximum of

$$\sum_{i=1}^{n}\sum_{k=1}^{K} \tau_{i,k}^{(t)}\, Q_{3,k}(\boldsymbol{\mu}_k|\boldsymbol{\Psi}^{(t-1)}) - \lambda_\mu \sum_{k=1}^{K}\sum_{d=1}^{p} |\mu_{k,d}|.$$

Despite this being differentiable with respect to $\mu_{k,d}$ only when $\mu_{k,d} \neq 0$, we can still set the derivative to zero and solve:

$$\sum_{i=1}^{n} \tau_{i,k}^{(t)}\, \frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(y_{i,d} - \mu_{k,d}) - \lambda_\mu\, \text{sign}\,(\mu_{k,d}) = 0 \tag{3.13}$$

while for the singular case where $\mu_{k,d} = 0$ the following inequality holds true:

$$\frac{\sum_{i=1}^{n} |\tau_{i,k}^{(t)}\, u_{i,k}^{(t)} y_{i,d}|}{\sigma_{k,d}^2} \leq \lambda_\mu. \tag{3.14}$$

Combining (3.13) and (3.14) the updating algorithm for $\boldsymbol{\mu}_k$ becomes

$$\boldsymbol{\mu}_k^{(t)} = \text{sign}(\tilde{\boldsymbol{\mu}}_k^{(t)}) \left(|\tilde{\boldsymbol{\mu}}_k^{(t)}| - \frac{\lambda_\mu}{\sum_{i=1}^{n} \tau_{i,k}^{(t)}\, u_{i,k}^{(t)}} \boldsymbol{\Sigma}_k^{(t)} \right)_+$$

where it can be seen that the unpenalised MLE of the mean, that is

$$\tilde{\boldsymbol{\mu}}_k^{(t)} = \frac{\sum_{i=1}^{n} \tau_{i,k}^{(t)} u_{i,k}^{(t)} \boldsymbol{y}_i}{\sum_{i=1}^{n} \tau_{i,k}^{(t)} u_{i,k}^{(t)}}$$

is shrunk towards zero by an amount that increases with λ_μ and is proportional to

the variance scaled by the precision factor. When λ_μ is sufficiently large

$$\frac{\sum_{i=1}^{n} |\tau_{i,k}\, u_{i,k}\, y_{i,d}|}{\boldsymbol{\sigma}_{k,d}} \leq \lambda_\mu$$

then $\mu_{k,d}$ collapses to zero thus making the d^{th} variable uninformative.

For completeness we shall add that when we adopt, as we do in later implementation, the adaptive version of the penalty function (3.5) with weight $\omega_{k,d}^{(\mu)}$, the updating algorithm for $\mu_{k,d}$ becomes

$$\mu_{k,d}^{(t)} = \text{sign}(\tilde{\mu}_{k,d}^{(t)}) \left(|\tilde{\mu}_{k,d}^{(t)}| - \frac{\lambda_\mu\, \omega_{k,d}^{(\mu)}\, \sigma_{k,d}^{(t)}}{\sum_{i=1}^{n} \tau_{i,k}^{(t)}\, u_{i,k}^{(t)}} \right)_+$$

Updating Step for $\boldsymbol{\Sigma}$

In an analogous way, the update for $\boldsymbol{\Sigma}$ is found by maximizing:

$$\sum_{i=1}^{n}\sum_{k=1}^{K} \tau_{i,k}^{(t)}\, Q_{3,k}(\boldsymbol{\Sigma}_k|\boldsymbol{\Psi}^{(t-1)}) - \lambda_\sigma \sum_{k=1}^{K}\sum_{d=1}^{p} |\log \sigma_{k,d}^2|$$

for $\boldsymbol{\Sigma}_k$ which is differentiable everywhere except for $\sigma_{k,d} = 1$. When $\sigma_{k,d} \neq 1$ its derivative with respect to $\sigma_{k,d}$ is

$$\sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2\,\sigma_{k,d}^2} + \frac{u_{i,k}^{(t)}(y_{i,d} - \mu_{k,d})^2}{2\,\sigma_{k,d}^4} \right) - \frac{\lambda_\sigma\ \text{sign}\,(\log \sigma_{k,d}^2)}{\sigma_{k,d}^2} \tag{3.15}$$

while for $\sigma_{k,d} = 1$ we have

$$\left| \sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2} + \frac{u_{i,k}^{(t)}(y_{i,d} - \mu_{k,d})^2}{2} \right) \right| \leq \lambda_\sigma. \tag{3.16}$$

The final update is obtained combining (3.15) and (3.16), which gives the following updates for $\boldsymbol{\sigma}_k^{2(t)}$,

$$\boldsymbol{\sigma}_k^{2(t)} = \left[\frac{\tilde{\boldsymbol{\sigma}}_k^{2(t)}}{1 + \lambda_\sigma\, \text{sign}(\boldsymbol{c_k} - b_k)/b_k} - 1 \right] \text{sign}(|b_k - \boldsymbol{c_k}| - \lambda_\sigma)_+ + 1$$

where $\tilde{\boldsymbol{\sigma}}_k^{2(t)} = \boldsymbol{c}_{\boldsymbol{k}}^t / b_k^t$ is the unpenalized maximum likelihood estimate of the standard deviation with

$$b_k^{(t)} = \sum_{i=1}^{n} \tau_{i,k}^{(t)}/2, \quad \boldsymbol{c}_k^{(t)} = \sum_{i=1}^{n} \tau_{i,k}^{(t)}\, u_{i,k}^{(t)} (\boldsymbol{y}_i - \boldsymbol{\mu}_k^{(t)})^2/2$$

It can be noted that, when λ_σ is sufficiently large

$$|b_k - c_{k,d}| \leq \lambda$$

then $\sigma_{k,d}$ is set to be one, effectively identifying the d^{th} variables noise and therefore non informative.

In this case too, since we use adaptive weights $\omega_{k,d}^{(\sigma)}$ to reduce the estimation bias introduced by the penalisation, we need to modify the updating algorithm for $\sigma_{k,d}^{2(t)}$ accordingly

$$\sigma_{k,d}^{2(t)} = \left[\frac{\tilde{\sigma}_{k,d}^{2(t)}}{1 + \lambda_\sigma\, \omega_{k,d}^{(\sigma)}\, \text{sign}(\, c_{k,d} - b_k)/b_k} - 1 \right] \text{sign}(|b_k - c_{k,d}| - \lambda_\sigma\, \omega_{k,d}^{(\sigma)}\,)_+ + 1.$$

All the main steps of the EM algorithm we have discussed are summarized in the Algorithm 3.1.

Algorithm 3.1 Expectation Maximization Algorithm

To find the MLE estimate $\boldsymbol{\Psi}$ that maximize the log likelihood of the penalised mixture of Student's t components $\log L_p(\boldsymbol{\Psi})$.

Initialize: At $t = 1$, Choose an initial setting for the parameters $\boldsymbol{\Psi}^0$

E Step: Evaluate the conditional expectation of the penalised log likelihood given $\boldsymbol{\Psi}^{t-1}$:

$$Q_p(\boldsymbol{w}|\boldsymbol{\Psi}^{(t-1)}) = E_{\boldsymbol{\Psi}^{(t-1)}}\{\log L_c(\boldsymbol{\Psi})|\, \boldsymbol{y}\} - h_{\boldsymbol{\lambda}}(\boldsymbol{\Theta})$$

M Step: Update estimates of $\boldsymbol{\Psi}$:

$$\boldsymbol{\Psi}^{(t)} = \arg\max_{\boldsymbol{\psi}} Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t-1)})$$

Check Convergence: If $|\log L_p^{(t)} - \log L_p^{(t-1)}| > 1$ return to E Step.

3.4 Model Selection

The EM algorithm we have described in the previous section allows us to efficiently search the parameter space to find the MLE of $\boldsymbol{\Psi}$, but takes K and $\boldsymbol{\lambda}$ as user-set parameters. Choosing the number of mixture components and the right level of penalisation is a critical decision that ultimately effects the quality of the inference we can draw from the model.

The clustering accuracy is clearly affected by the number of clusters that we assume are present in the data. Setting a too low or too high value K, would force the algorithm to either aggregate different groups or separate similar observations respectively. The penalty vector, on the other hand, controls the variable selection process since a variable is retained as informative only if its density parameters are above a deterministic threshold which is a function of $\boldsymbol{\lambda} = (\lambda_\mu, \lambda_\sigma)$. For any given level of the regularisation there will be a set of selected variables, $S_{\boldsymbol{\lambda}} \subseteq \{1, \ldots, p\}$ having cardinality m, which is an estimate of the true structure of the data.

Both the optimal number of mixture components and level of penalisation can

be found by exploring a finite number of solutions and adopt a model selection criteria to choose the best. One approach to model selection is to pick the candidate model with the highest probability given the data. This idea can be formalised inside a Bayesian framework, involving prior probabilities on candidate models along with prior densities on all parameter vectors in the models. The Bayesian procedure requires to select that model which is a posteriori most likely. The best solution can be identified by calculating the posterior probability of each alternative model and selecting the one with the biggest posterior probability.

The Bayesian information criterion (BIC) of Schwarz (1978) approximates the Bayesian posterior probability of the model. It takes the form of a penalised log-likelihood function where the penalty is equal to the logarithm of the sample size times the number of estimated parameters in the model.

In our penalised likelihood framework, we use the modified version proposed of the BIC, as suggested by (Pan and Shen, 2007),

$$\text{BIC} = -2 \log L_p(\boldsymbol{\Psi}) + r \log(n)$$

where n is the number of samples and $r = g - 1 + 2\,g\,m + g - q$ is the effective number of parameters once the $q = p - m$ non-informative variables are excluded from the model. The modification is introduced to favour the more parsimonious models that can be found through penalisation.

The Akaike information Criterion (AIC) (Akaike, 1973)

$$\text{AIC} = -2 \log L_p(\boldsymbol{\Psi}) + 2\,r$$

is an alternative criterion that conforms to the same principle of the BIC, but does not take into consideration the number of observations n. Based on preliminary tests, we found that the BIC and AIC do not give significantly different indications, and therefore we adopted the BIC as sole criterion.

However, both criteria do not always lead to the correct choice of the best model (Baek and McLachlan, 2011). In our experience, when m is very small compared

to p and when the densities are fat tailed we find that information based criteria still prefer models which are too complex as some degree of over-fitting still takes place.

3.4.1 Resampling Strategies for Stability Selection

To overcome some of the shortcomings of the BIC, we propose a subsampling approach which is similar to the stability selection procedure originally developed for variable selection in penalised regressions models by Meinshausen and Bühlmann (2010). This procedure enhances the model selection, but also provides a way to rank the selected variables.

Initially we assume that the number of mixture components is fixed. For a given K, we are interested in selecting an optimal set of informative variables, which should be ranked in decreasing order of importance. We search for a value of $\boldsymbol{\lambda}$ that minimises the modified BIC criterion, and call this optimal value $\boldsymbol{\lambda}^*$. This search can be carried out using a grid of candidate points. Then, B subsamples of the data are randomly drawn, $\{\boldsymbol{y}^{(b)}\}_{b=1}^{B}$, all having size $[n/2]$. For each random sub-sample $\boldsymbol{y}^{(b)}$, we fit the penalised mixture model using the EM algorithm with the given number of components K and regularisation $\boldsymbol{\lambda}^*$. The set of variables selected in each sub-sample is called $S_{\boldsymbol{\lambda}^*}^{(b)}$. An indicator variable $I_d(S_{\boldsymbol{\lambda}^*}^{(b)})$ is introduced which equals 1 if the variable d has been flagged as informative for $\boldsymbol{y}^{(b)}$, and zero otherwise. The selection probability for gene d is then estimated as

$$\hat{\pi}_d = \frac{\sum_{b=1}^{B} I_d(S_{\boldsymbol{\lambda}^*}^{(b)})}{B}, \quad d = 1, 2, \ldots, p. \tag{3.17}$$

It can be noted that, whereas a single model fit obtained with the EM algorithm using the whole data set would only provide a binary indicator labelling each variable as informative or not, the selection probabilities provide a useful metric to assess the relative importance of each feature both for clustering as well as for ranking purposes. All the variables having a sufficiently high selection probability are then deemed informative. A threshold on $\hat{\pi}_d$ could be selected to control the number of false positives, as in Meinshausen and Bühlmann (2010), although little

theoretical developments are available yet.

Apart from enabling to rank variables, the resampling approach provides the means to improve upon the model selection process. In order to estimate the correct number of mixture components, K, it is common routine to compare a series of models, each one having an increasing number of components, say from 2 to $K_{\max}$, and select the model with the smallest BIC. However, when the ratio of non-informative over informative variables is high, we have found that the modified BIC still tends to overestimate the number of clusters. We address this issue by proposing the following two-step procedure. For each one of the $k_{max} - 1$ models being compared, we carry out the subsampling procedure as described above, and collect in a set of cardinality $\tilde{m}$ the informative variables selected by all models over all subsamples. In a second step, we re-fit all competing models, but instead of using all the p features, we use only the $\tilde{m}$ informative features, where $\tilde{m}$ is usually much smaller than p. The best model is the one that minimises the BIC, as usual. By initially removing the non-informative variables, and therefore the amount of noise, this simple approach reduces the bias towards selecting more complex models, and improves upon the selection of the number of components, as shown in Section 3.5.

3.5 Experimental Results

Having presented the main properties of the penalised Student's t mixture model and having described the EM algorithm to estimate the unknown parameters, we now want to assess how well the model performs in practice, using simulated data.

We first verify that the model selection procedure introduced in section 3.4 can effectively help us to identify the correct number of clusters K and the appropriate level of penalisation $(\lambda_\mu, \lambda_\sigma)$. Moreover, we illustrate how, in the typical noisy scenario, the proposed resampling technique can improve the BIC driven model selection process.

We then focus our attention on the actual clustering and variable selection performance of the model. We simulate several different scenarios and compare

the accuracy of the model against popular competing clustering algorithms.

Finally, to stress test the model, we simulate settings close to the real life problems such as those presented in chapter 1, and find that the results are in line with our expectations.

Data Simulation Procedures

In the following discussion we consider several different scenarios and for each scenario we generate multiple datasets which are randomly sampled from a mixture of multivariate Student's t distributions, as described by the following generative model:

$$\boldsymbol{y} \sim \sum_{k=1}^{K} w_k \, f_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \nu_k).$$

Each dataset $\boldsymbol{y}$ is an $(n \times p)$ matrix with m informative variables whose parameters are different in distinct components and $q = p - m$ non-informative variables that share the same parameters across all components.

The exact settings of each scenario depend on the particular aspect of the model we want to test at that point. In all cases, though, to be as close as possible to real life situations, the sample size is kept relatively small, $n << p$, and the number of uninformative variables is always taken to be much higher than the informative variables, $m << q$.

Performance Measures

The mixture model is evaluated in each scenario for its clustering and variable selection accuracy. For this purpose we employ three performance indicators.

Once the EM algorithm has converged, we assign each observation $\boldsymbol{y}_i$ to the cluster with the highest posterior probability $\tau_{i,k}$. The clustering performance, is then evaluated using the Adjusted Rand Index (`ARI`) (Hubert and Arabie, 1985). This index measures the proximity between any two alternative partitions of the samples, and we use it to establish how close the estimated clusters assignments are from the ground truth, where a value 0 denotes random assignment and 1

perfect matching.

We assess the accuracy of the model in identifying the informative variables by computing the sensitivity index

$$\texttt{Sens} = \frac{\#\text{ true positive}}{(\ \#\text{ true positive} + \#\text{ false negative})}$$

which is the ratio of the informative variables that have been correctly selected by the algorithm (true positives) over the total number of informative variables. Alternatively we can quote the False Negative Rate (FNR) which is defined as $\texttt{FNR} = 1 - \texttt{Sens}$.

Similarly, in order to measure how well the model is able to exclude all the uninformative variables, we compute the specificity index

$$\texttt{Spec} = \frac{\#\text{ true negative}}{(\ \#\text{ false positive} + \#\text{ true negative})}$$

which is the ratio between the number of negative variables that have been excluded from the model over the total number of non informative variables. The False Positive Rate (FPR) would give us the same information and it is simply defined as $\texttt{FPR} = 1 - \texttt{Spec}$

Competing Models

In the various scenarios considered, we compare the performance of the penalised t-Student mixture model (PTM) against competing clustering methods such as those we reviewed in section 2.4.1. This approach allows us to better appreciate the peculiar properties of the proposed model.

Since we are interested in justifying the adopted joint L_1-norm penalty function, it seems obvious, first of all, to consider the alternative models: $\texttt{PTM}_\mu$ that penalises only the mean parameter, $\texttt{PTM}_\sigma$ that penalises only the variance and TM that does not perform any variable selection.

To highlight the benefit of robust modelling, we benchmark the performance of the penalised mixture of Student's t distribution against a penalised mixture of

Gaussian distributions (PGM) as originally implemented by Xie et al. (2008a). For symmetry we consider the Gaussian version of the different penalisation options: $\texttt{PGM}_\mu$, $\texttt{PGM}_\sigma$ and GM.

Since both PTM and PGM follow a model-based clustering approach, in some scenarios, for completeness, we also evaluate the sparse K-means algorithm (PKM) (Witten and Tibshirani, 2010), which uses a lasso-type penalty to select the relevant variables and obtain a sparse partitioning clustering.

3.5.1 Variable Selection Demonstration

We shall begin the discussion of the experimental results, by illustrating the procedure we implement in order to find the optimal level of penalisation λ_μ and λ_σ. As mentioned earlier, we will fit a series of models using different combination of λ_μ and λ_σ and chose the most promising model according to the modified Bayesian Information Criterion (Pan and Shen, 2007).

For the moment, we assume perfect insight into the number of components and run the EM algorithm with the right value K fixed. The exact details of the remaining simulation settings are described in section 3.5.5, scenario 1 and scenario 2. They have been chosen such that they are representative of two extreme opposite situations: one where variables are sampled from a t distribution with high degrees of freedom, the other where variables are sampled from a t distribution with low degrees of freedom.

Figure 3.1 and 3.2 show the results of the grid search over all possible combinations of λ_μ and λ_σ for values ranging between 0 and 20. The contour plots confirm that the minimum BIC point on the grid corresponds also to the minimum variable selection error, $\texttt{TOTE} = (\texttt{FPR} + \texttt{FNR})$, that is the optimal point where the false negatives and false positives rates are lowest. Not surprisingly then, the combination of λ_μ and λ_σ that minimizes the BIC index is also the best model that achieves the highest percentage of correct clustering assignment, Right.

Figure 3.2, in particular, confirms that the proposed penalised mixture of t distributions can fit better the heavy tailed data and has a significantly higher

likelihood than the penalised Gaussian mixture. Note how the contour plots on the right column, $\texttt{PTM}_{\mu,\sigma}$, have brighter colours, i.e. dominate the plots on the left column, $\texttt{PGM}_{\mu,\sigma}$. This result can be explained by the fact that a better variable selection ultimately leads to an higher clustering accuracy.

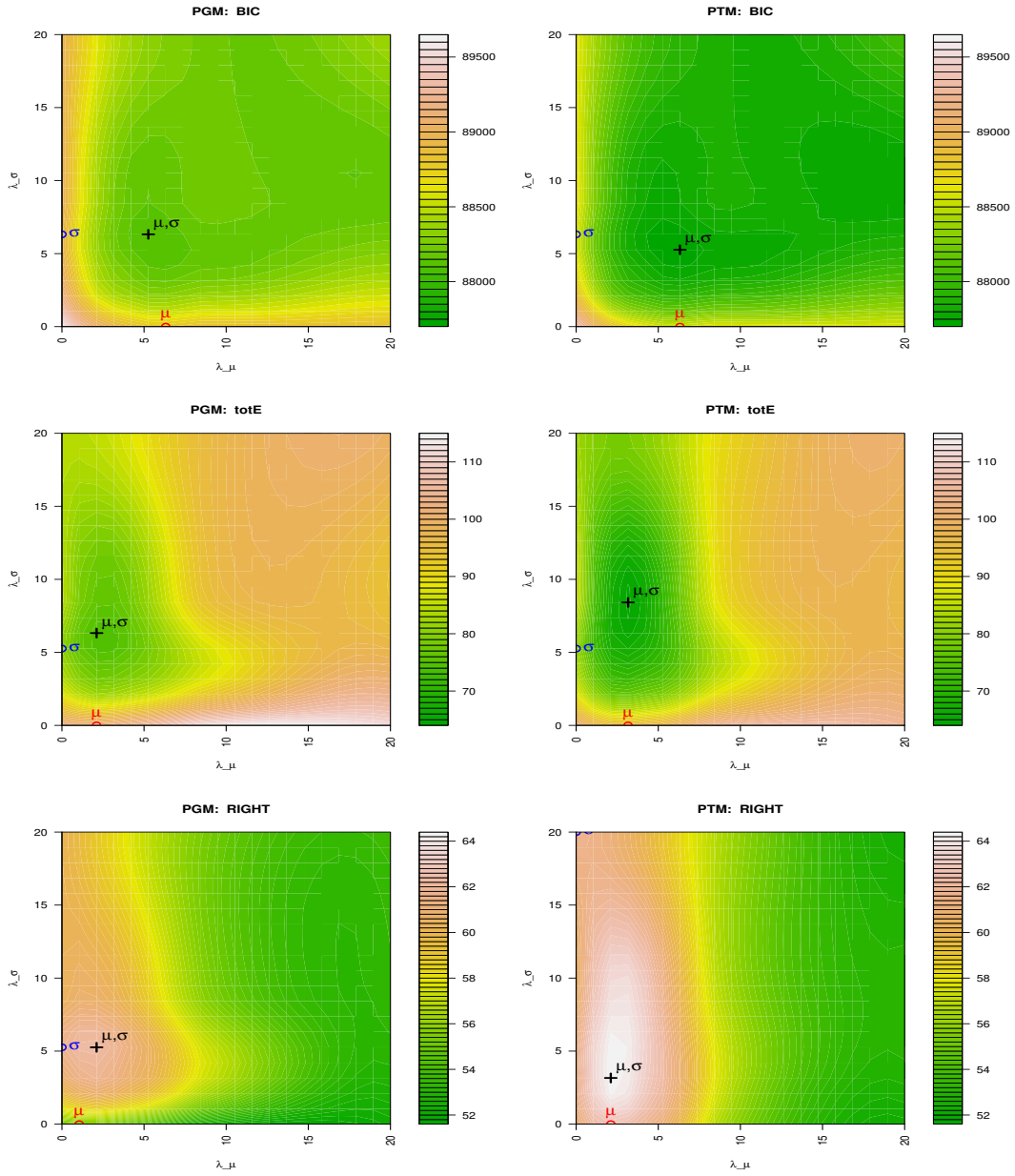

Figure 3.1: Scenario 1. Model selection with sample data generated from high degrees of freedom t mixture. Each quadrant represents the grid of the combinations of λ_μ and λ_σ. In the top row of plots the crosses indicate the lowest BIC level achieved by the different penalty functions: black cross for the best joint penalty combination λ_μ and λ_σ, red cross for the best single λ_μ penalisation, i.e. keeping $\lambda_\sigma = 0$, and blue cross for the best λ_σ. In the middle row we report the total variable selection error, $\texttt{TOTE} = (\texttt{FPR} + \texttt{FNR})$. The bottom row shows the percentage of observations that have been assigned to the correct cluster, `Right`. Observing the alignment of the crosses from top to bottom plot, it is evident that the minimum BIC point corresponds also the minimum variable selection error and indicates the model with the highest clustering accuracy.

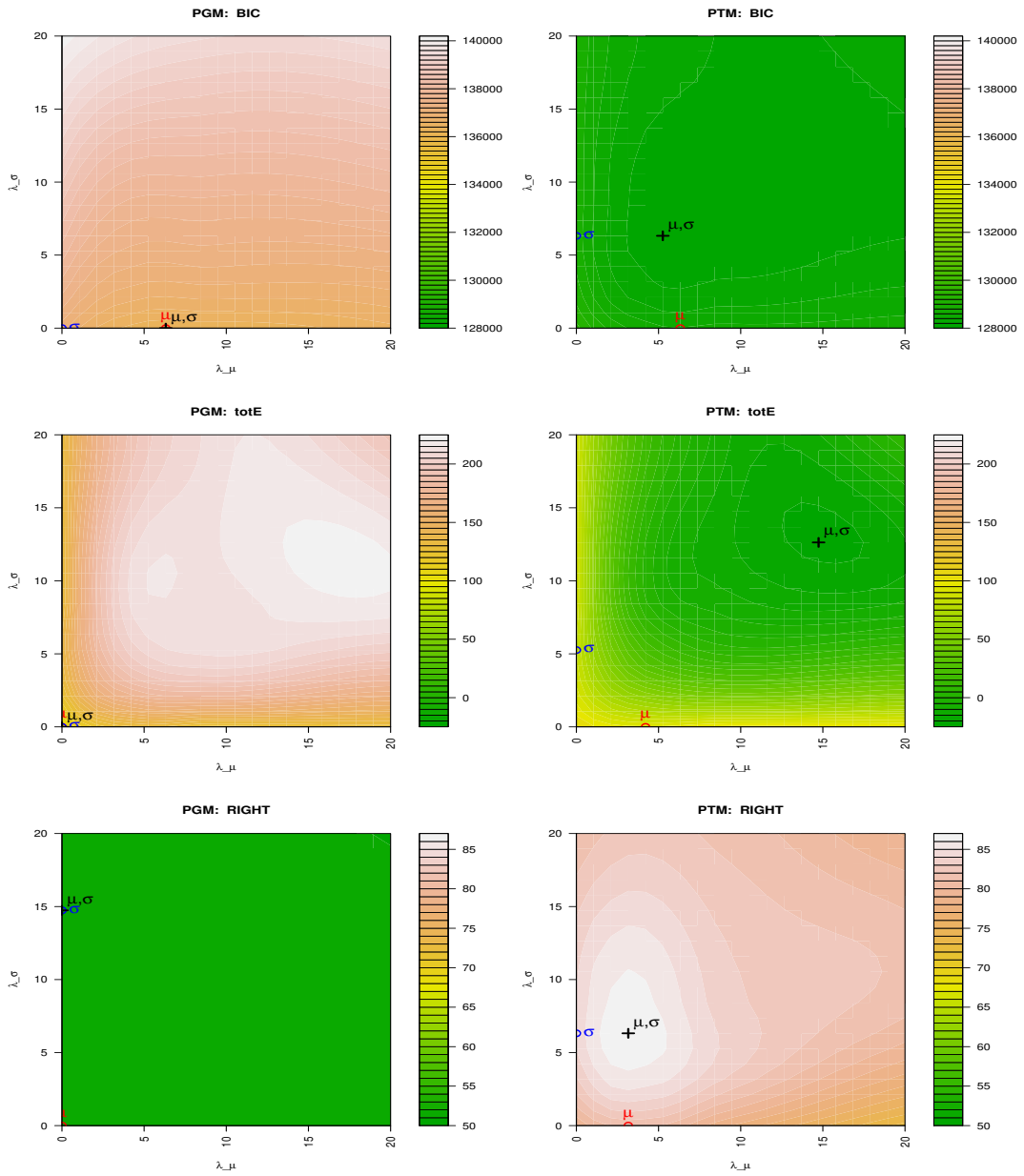

Figure 3.2: Scenario 2, Model selection when variables are sampled from a t mixture with low degrees of freedom. Note how the clearly more vivid colours of the plots in the right column indicate a significant outperformance of `PTM` relatively to `PGM`. A lower BIC and lower variable selection error, `TOTE`, correspond to a considerable higher clustering accuracy, `RIGHT`. We should also point out that the guidance offered by the BIC is still reliable for `PTM`, but less so for `PGM`, in this case the lowest BIC point does not correspond to the highest percentage of correct assignment.

ROC curves

We have shown how variable selection improves the clustering performance in the presence of noisy non informative data. Here we illustrate how we have been able to correctly identify the informative variables by maximizing the true positive ratio `TPR` and minimizing the error of retaining non informative variables, false positive ratio `FPR`.

In Figure 3.3 and 3.4 we show separate Receiver Operating Characteristic (ROC) curves generated by gradually increasing the level of penalisation λ_μ and λ_σ imposed on the location and dispersion parameter respectively. In the first scenario, Figure 3.3, we simulate data sampled from a mixture of approximately Gaussian distributions. We observe that the performance of the two models, `PGM` and `PTM`, is fairly similar and quite accurate. We notice that, as we increase the level of penalisation, the ratio of false positives variables drops much faster than the true positive variables. We are able to effectively filter out the noise without degrading the signal.

In the second scenario, Figure 3.4, data have been sampled from a mixture of long tail t distributions. In this case separating the noise from the informative variables is more difficult. Nonetheless we observe a noticeable overperformance of the `PTM` model as it can sustain an higher `TPR` while penalisation is increased. The results are justified by the fact that `PTM` can recognise and filter out the non informative ones better than the Gaussian mixture models can. In particular we should highlight how the performance of `PGM` is less robust and more disperse with few instances of below par results.

For completeness we should mention that in extreme scenarios, where σ parameters of the different components are very similar while the noise variables have a very long tail, both models fail to accurately isolate the truly informative variables. In Figure 3.5, we observe that even if `PTM` shows a slightly better performance, it still erroneously excludes the informative variables before excluding the noise ones. The reason is that when fitting the mixture of distributions, both models find spurious clusters in the very long tailed noise variables while the signal

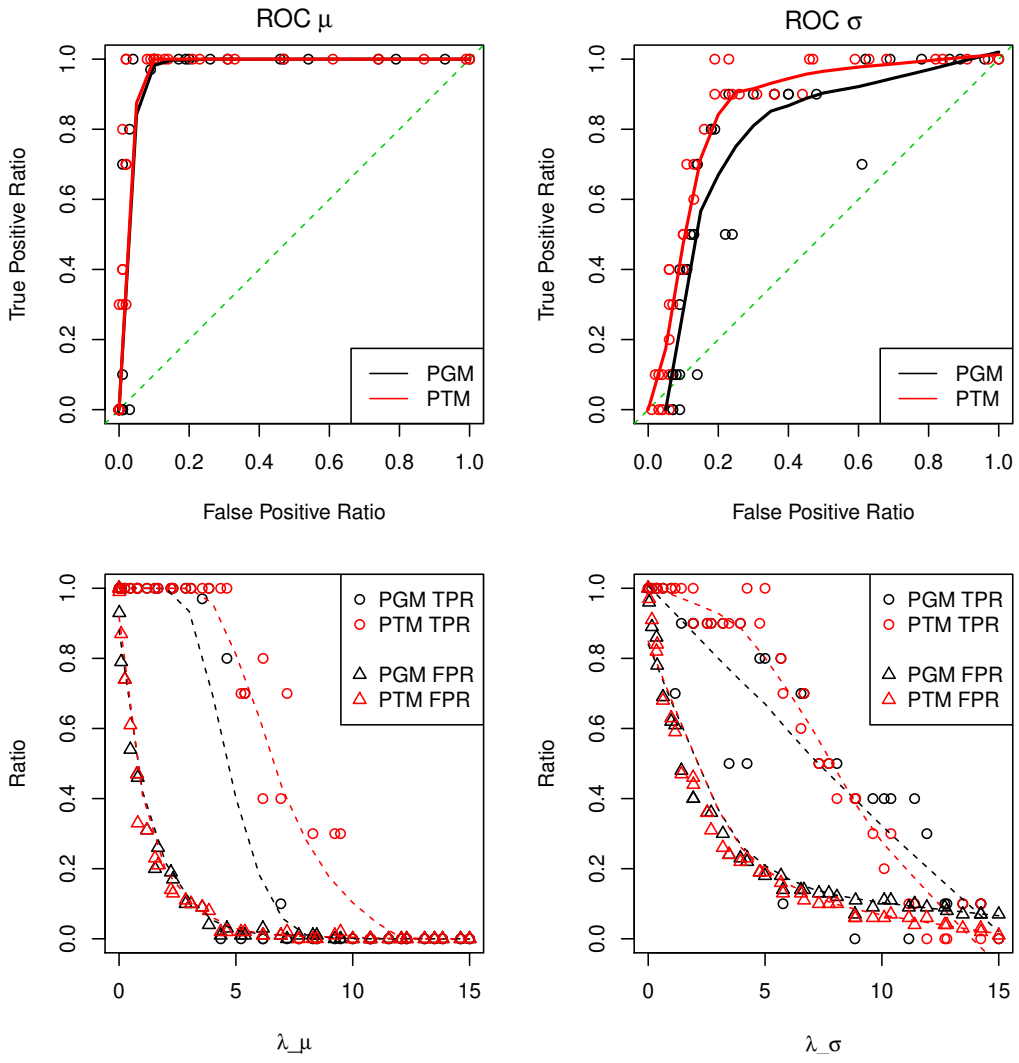

Figure 3.3: ROC curves from Scenario 1 where noise variables have high degrees of freedom. No noticeable difference between the two models, `PGM` and `PTM`, they both can correctly identify the non informative variables without penalising the informative variables too. The slightly better performance of `PTM` is due to its robustness to noise.

in the informative variables is very low. In this limit case the informative variables are dropped by the model before the noise ones and the ROC curves become convex.

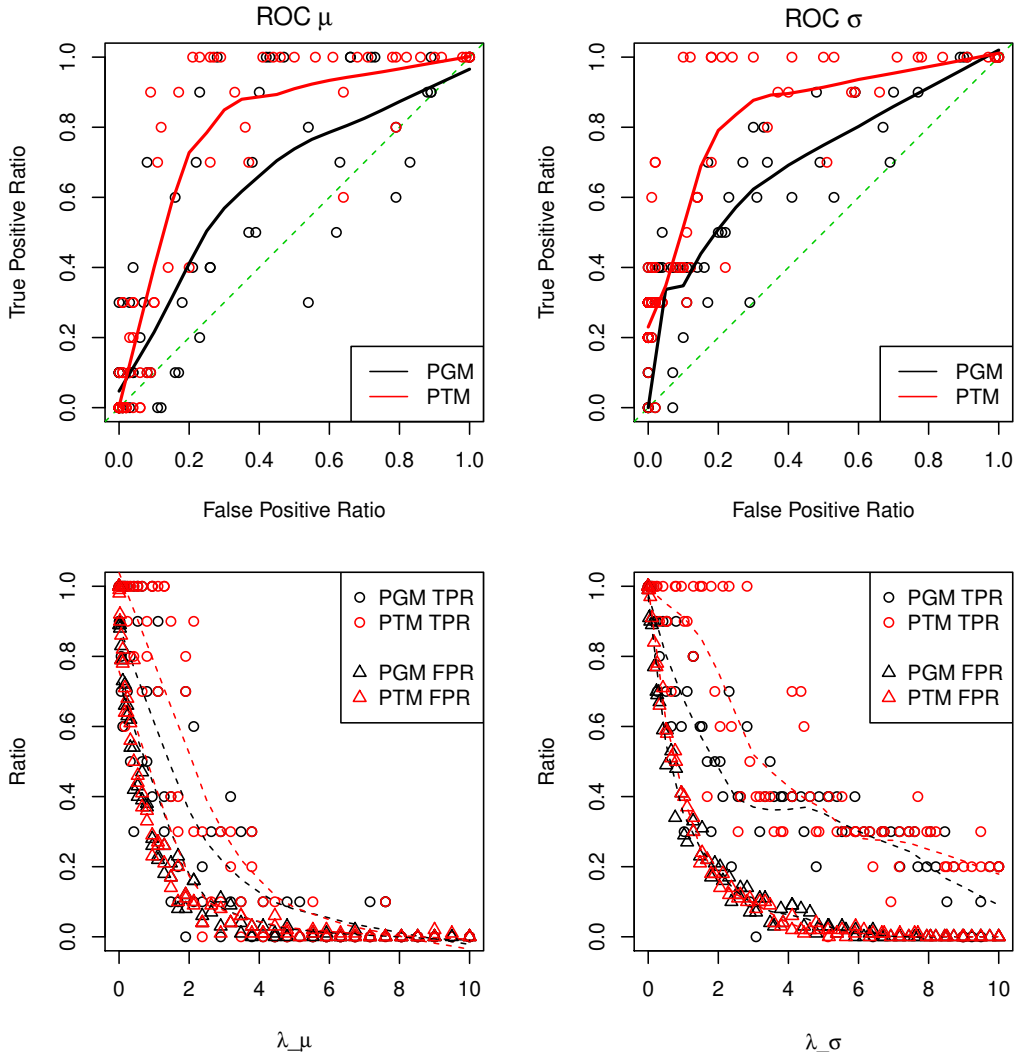

Figure 3.4: ROC curves from Scenario 2. All variables are generated from long tailed Student's t density functions. The advantage of the penalized Student's t Mixture, `PTM`, is more significant. The penalty imposed by `PGM` on $\boldsymbol{\mu}$ does not always help to isolate the non-informative variables and therefore tend to exclude the same proportion of informative and noise variables.

3.5.2 Subsampling Strategy for Variable Selection

Stability selection is an effective subsampling method that can address the problem of proper regularization and help us improve the variable selection process. It stresses the robustness of the proposed model by performing, for a significant number of iterations, a random subsampling of the observations and refitting the model. The relative importance of each variable is gauged by analyzing its selection

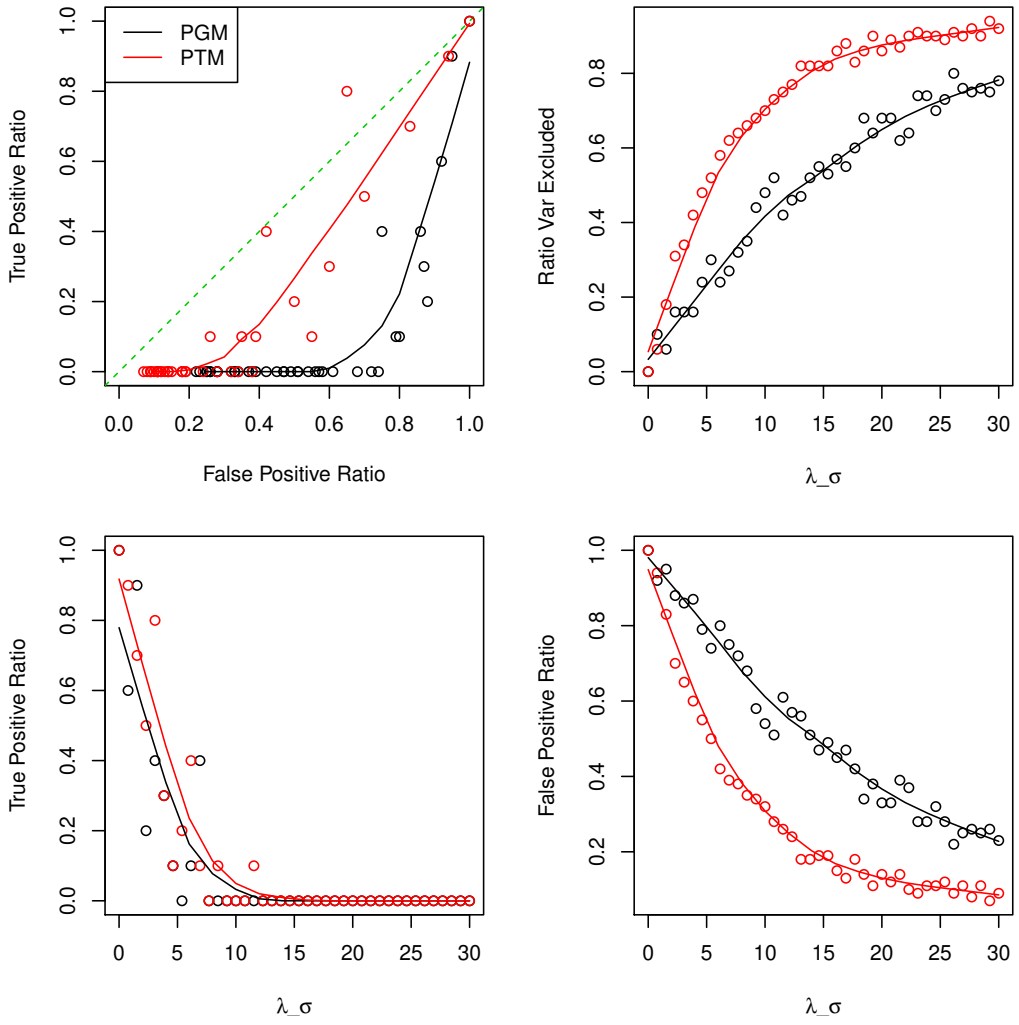

Figure 3.5: ROC curves in extreme Scenario where σ parameters of the different components are very similar while the noise variables have a very long tail. In this case both models are mislead to believe there is more signal in the noise variable and spuriously isolate different clusters in the non informative variables. The informative variables are penalised more than the noise variables.

frequency and using this information we refine the variable selection process. While bootstrapping would allow for sampling with replacement it behaves very similarly as noted by Meinshausen and Bühlmann (2010).

To illustrate the point, we generate multiple random datasets from a mixture of multivariate Student's t distributions. Each dataset contains $n = 200$ observations and for each observation we have $m = 20$ informative variables sampled from a mixture of three clusters, $\{A, B, C\}$, with mixing proportions: $\{w_A = 0.3, w_B =$

$0.2, w_C = 0.5\}$. The exact parameters of the t Student density functions that generate the informative variables are: $\boldsymbol{\Theta}_A = \{\boldsymbol{\mu}_A = -3, \boldsymbol{\sigma}_A = 3, \nu_A = 5\}$, $\boldsymbol{\Theta}_B = \{\boldsymbol{\mu}_B = 4, \boldsymbol{\sigma}_B = 4, \nu_B = 6\}$ and $\boldsymbol{\Theta}_C = \{\boldsymbol{\mu}_C = 6, \boldsymbol{\sigma}_C = 2, \nu_C = 8\}$. The $q = 2000$ non-informative variables, instead, have been sampled from a single fairly long tailed t distribution: $\boldsymbol{\Theta}_Q = \{\boldsymbol{\mu}_Q = 0, \boldsymbol{\sigma}_Q = 1, \nu_Q = 4\}$.

In this instance, for each level of penalisation $\boldsymbol{\lambda}$ between 0 and 14, we iterate 100 times the random subsampling of size $[n/2]$ and fix the probability selection threshold at $\tilde{\pi} = 0.7$, as suggested by Meinshausen and Bühlmann (2010)

In Figure 3.6 we can appreciate the impact of higher penalty on the selection probability of the non informative variables. The plots shows that the selection probability of the non informative variables quickly drops as we increase $\boldsymbol{\lambda}$. The informative variables, on the other hand, are retained by almost every subsample iteration, even as we impose a stronger penalization.

In Figure 3.7, we slice the stability paths at the optimal level of penalisation λ^*_μ and λ^*_σ as indicated by the BIC criterion. We can see there that the informative variables, the first 20 variables of each sampled observation, are very clearly separated from the rest of non informative variables. The hard threshold line identifies exactly the variables that should be retained.

A more accurate variable selection process, invariably leads to a more accurate clustering performance as we can see from table 3.1. The adjusted rand index increases because we have been able to further improve the specificity of the model and exclude more noisy variables.

Resampling	`ARI`	`Sens`	`Spec`
No	93 (5)	100 (1)	87 (8)
Yes	98 (2)	100 (0)	99 (1)

Table 3.1: Clustering and variable selection performance. The resampling routine improves the specificity of the variable selection process and consequently achieves a better clustering accuracy.

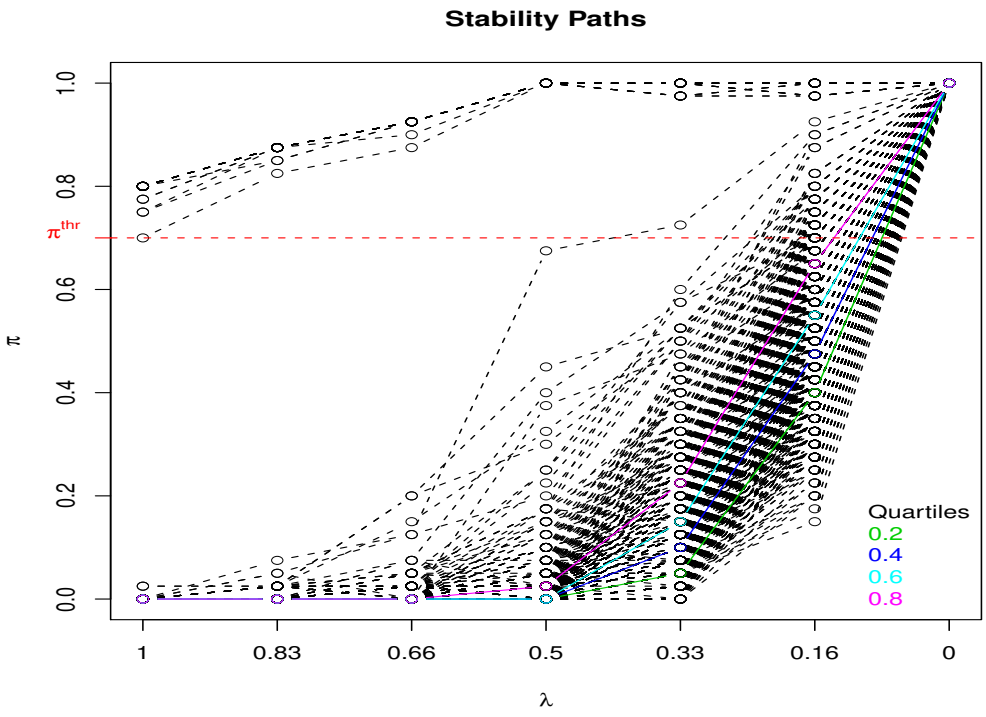

Figure 3.6: Stability Paths. Effect of increasing penalisation λ on the selection probability of each variable. Only the informative variable are constantly selected in every subsample iteration. Note that the range of penalisation has been rescaled between 0 (no penalisation) and 1 (maximum penalisation). Red dotted line corresponds to the selection probability threshold.

3.5.3 Model Selection: Number of Clusters

In the previous section, we have illustrated how we are able to select the informative variables by finding an optimal combination of λ_μ and λ_σ, and also shown that a straightforward resampling strategy can make the selection process more robust and improve the performance of the model. Here we describe the procedure we follow to identify the number of clusters in the sampled data. In this case too, we find evidence that stability selection can significantly help us to find the right answer.

We generate a single dataset using the same simulation settings of the previous section, that is we sample from mixture of three long tailed Student's t distributions. We then fit a different model for each tested value $K = 2, \ldots, 5$ where for

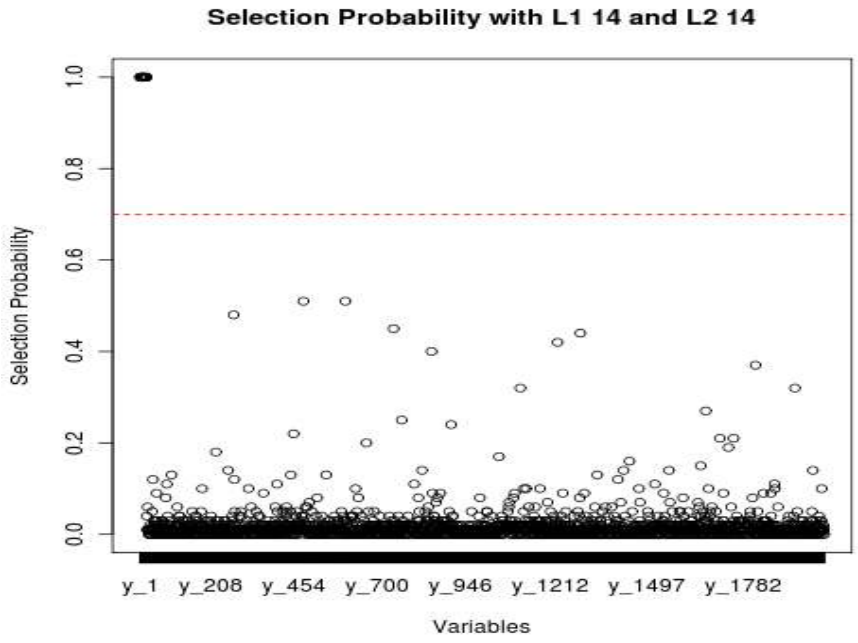

Figure 3.7: A different way of slicing the stability paths that shows how the first 20 variables, the only ones informative, are always selected in every random subsampling. For a reasonably high selection threshold, $\tilde{\pi} = 0.7$ the right variables are selected.

each K we still perform a separate grid search to identify the best level of penalisation. Note anyway that, as we can see in Figure 3.8, the optimal combination of λ_μ^* and λ_σ^* is generally stable and does not depend on the number of clusters.

For each model we compute the log-likelihood `LLIK`, the `AIC` and `BIC` score, which are reported in table 3.2. The results show that, in this case, all three criteria agree and correctly indicate $K = 3$ as the most likely hypothesis.

# Clusters	λ_μ	λ_σ	`LLIK`	`AIC`	`BIC`
2	6	8	-571156	1142513	1142842
3	6	10	-571113	1142409	1142709
4	8	10	-571150	1142474	1142761
5	10	10	-571137	1142456	1142756

Table 3.2: Model Selection. Log likelihood, Akaike and Bayesian Information score of each of the fitted model for $K = 2, \ldots, 5$. Note that the true number of components is three.

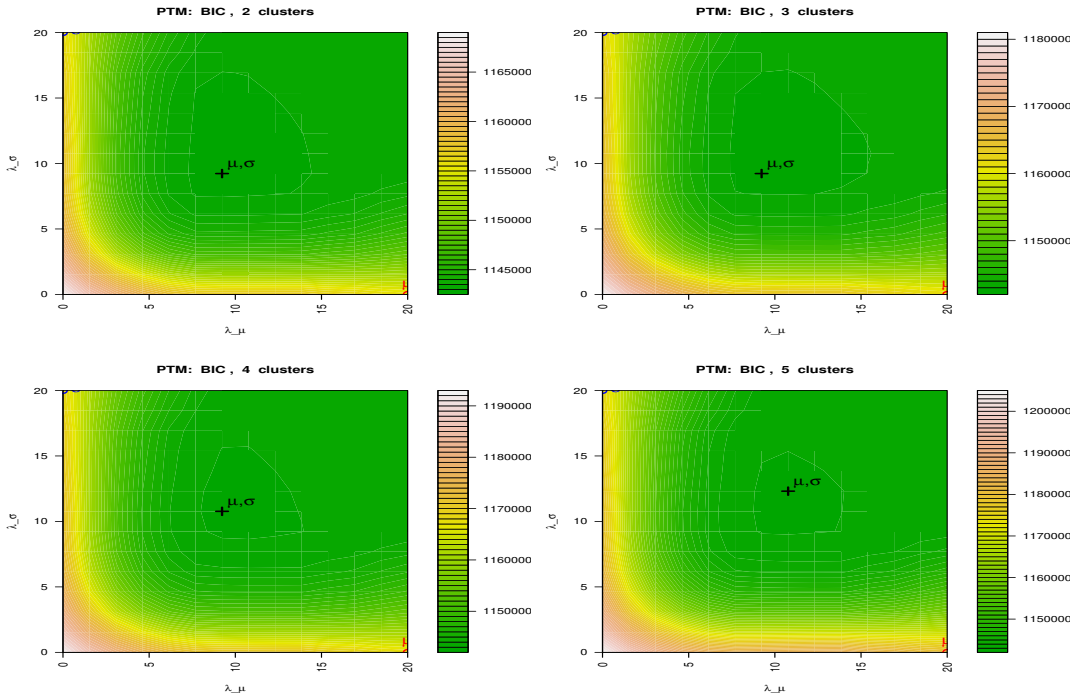

Figure 3.8: Model Selection: BIC contour plots for all combinations of penalisation λ_μ and λ_σ, under different assumptions about the number of clusters, $K = 2, \ldots, 5$.

3.5.4 Resampling Strategy for Model Selection

In this section we want to illustrate how the subsampling strategy can contribute to increase the model selection accuracy.

To produce a stronger evidence of the relative difference between the two methods, we need to simulate slightly more realistic scenarios where the overlap between the components is higher and the tails of the distributions are longer. As before, the dataset contains $n = 200$ observations and each observation is made of $m = 20$ informative variables sampled from a mixture of three clusters,$\{A, B, C\}$, with mixing proportions: $\{w_A = 0.3, w_B = 0.2, w_C = 0.5\}$. The density functions that generate the informative variables have now lower degrees of freedom and are centered closer together: $\boldsymbol{\Theta}_A = \{\boldsymbol{\mu}_A = -3, \boldsymbol{\sigma}_A = 3, \nu_A = 3\}$, $\boldsymbol{\Theta}_B = \{\boldsymbol{\mu}_B = 0, \boldsymbol{\sigma}_B = 4, \nu_B = 3\}$ and $\boldsymbol{\Theta}_C = \{\boldsymbol{\mu}_C = 3, \boldsymbol{\sigma}_C = 2, \nu_C = 3\}$. The $q = 200$ non-informative variables are sampled from a single fairly long tailed t distribution:

$\boldsymbol{\Theta}_Q = \{\boldsymbol{\mu}_Q = 0, \boldsymbol{\sigma}_Q = 1, \nu_D = 3\}$. The level of penalisation $\boldsymbol{\lambda}$, as before, ranges between 0 and 14. The subsampling routine is iterated 100 times, using $n/2$ observations each time, and only the variables whose selection probability is above $\tilde{\pi} = 0.7$ are retained.

In tables 3.3 and 3.4 we report the results of the simulation using the two different methodologies, one which does not execute a subsampling routine and one which does. We can see that, in the first case, the guidance provided by the BIC criteria is misleading because it suggests an higher number of components than there are in reality. In the second case, by implementing a more accurate variable selection procedure using the resampling routine, we are able to exclude all the noise variables and unmask the true structure of the data. In fact, this time, the BIC index correctly identifies $K = 3$ as the best model. Note also how the clustering performance greatly benefits from the resampling step as demonstrated by the better `ARI` score.

# Clusters	λ_μ	λ_σ	`LLIK`	`AIC`	`BIC`	`ARI`	`Sens`	`Spec`
2	8	8	-570310	1140923	1141421	61	100	95
3	14	8	-569606	1139729	1140580	86	100	93
4	14	8	-569109	1138986	1140253	75	100	86
5	12	10	-569246	1139083	1140056	75	100	90

Table 3.3: **No Bootstrap.** We test all possible assumptions , $K = 2, \ldots, 5$, and perform a BIC grid search to find optimal level of penalisation, column λ_μ and λ_σ. Looking at the log likelihood and BIC score of all models, we would be mislead to believe that the $K = 5$ is the most likely model while a mixture of three clusters is the true representation. Note in fatc how the Rand Index is highest at $K = 3$.

3.5.5 Penalised t Mixture Vs Penalised Gaussian Mixture

In this section we analyze the performance of the proposed mixture of Student's t model implementing adaptive L_1-norm penalization either on mean parameters, $\texttt{PTM}_\mu$, on variance, $\texttt{PTM}_\sigma$, or jointly on μ and σ, $\texttt{PTM}_{\mu,\sigma}$. To better appreciate the specific qualities of the penalised t mixture we benchmark their performance against penalised Gaussian mixture models $\texttt{PGM}_\mu$, $\texttt{PGM}_\sigma$ and $\texttt{PGM}_{\mu,\sigma}$ as implemented by Pan

# Clusters	λ_μ	λ_σ	LLIK	AIC	BIC	ARI	Sens	Spec
2	8	8	-3906	7979	8253	61	100	100
3	14	8	-3662	7575	7988	99	100	100
4	14	8	-3619	7573	8124	95	100	100
5	12	10	-3574	7567	8256	71	100	100

Table 3.4: **With Boostrap**. Maintaining the same level of penalisation, we now run the boostrap routine which helps us to eliminate any residual variable selection error, column Sens and Spec. The BIC criteria now correctly identifies $K = 3$ as the most likely model which also achieves almost perfect clustering performance.

and Shen (2007) and Xie et al. (2008a). To remark the importance of variable selection, we also report the results of the unpenalised mixtures of Gaussians, GM, and mixtures of t, TM, as described by McLachlan and Peel (2000).

Looking at the results, we find that, in scenario 1, penalized Gaussian and t mixtures models perform similarly, which is a reasonable conclusion since the marginal densities of informative and non-informative variables belong to the Gaussian family. In scenario 2, we find evidence that t mixtures significantly overperforms Gaussians models as the clusters' densities deviate from normality. The difference is explained by a more accurate variable selection of the t mixture. This is evident also in scenario 3 where all the components of the mixture have approximately Gaussian density but $\texttt{PGM}_{\mu,\sigma}$ still underperforms $\texttt{PTM}_{\mu,\sigma}$ because the non-informative variables are t distributed with long tails. In scenario 4 we generate a bigger data sample where the number of variables is significantly higher than the number of observations with only a small percentage of informative variables. We find confirmation of the previous results and note that $\texttt{PTM}_{\mu,\sigma}$ is able to select only the useful variables because it can fit better their long-tail distributions.

Simulation Scenarios

For the benefit of the visual representation of the results, we assume throughout that there are only two components, $K = 2$, nominally A and B having the same mixing proportion $w_A = w_B = 1/2$. The different scenarios are then generated using different settings for the components' densities $\boldsymbol{\Theta}$.

In the first instance we vary the degrees of freedom, ν, as we investigate how the clustering performance responds to changes in the tail shape of the variables' distribution. We consider two opposite scenarios.

Scenario 1: High Degrees of Freedom. We assume $\nu = 40$ so that the density of the simulated variables approximates the Gaussian density. We generate $n = 200$ samples each one with only two informative variables, $m = 2$, coming from either cluster A which has parameters $\boldsymbol{\Theta}_A = \{\boldsymbol{\mu}_A = -2, \boldsymbol{\sigma}_A = 3, \nu_A = 40\}$ or cluster B which has parameters $\boldsymbol{\Theta}_B = \{\boldsymbol{\mu}_B = 4, \boldsymbol{\sigma}_B = 4, \nu_B = 40\}$. The remaining variables, $q = 98$, are not useful for clustering and are sampled from a single distribution with parameters $\boldsymbol{\Theta}_q = \{\boldsymbol{\mu}_q = 0, \boldsymbol{\sigma}_q = 1, \nu_q = 40\}$. In Figure 3.9 we show the scatterplot of the sampled informative variables and the 3D surface of the mixing density functions.

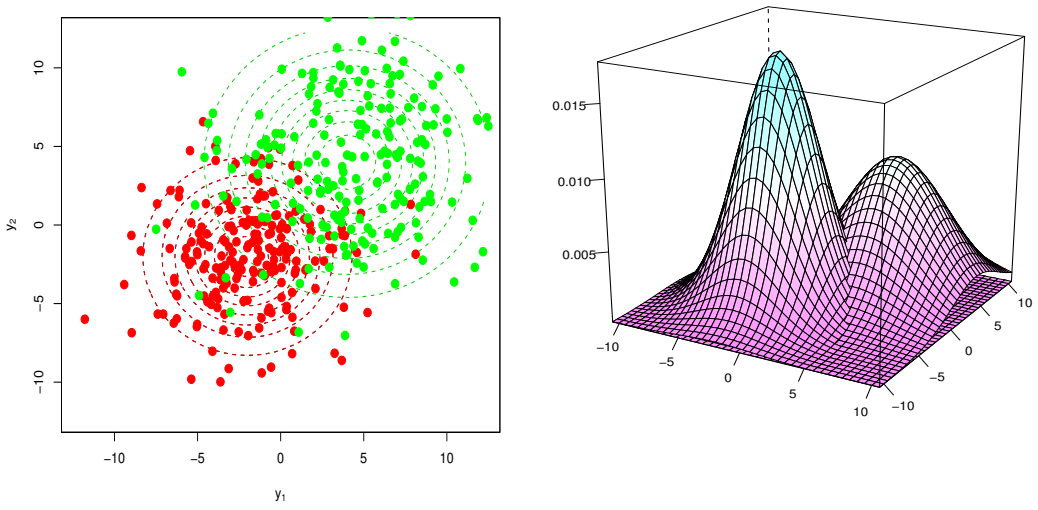

Figure 3.9: Scenario 1: High degrees of freedom. The two components of the mixture, A in red and B in green, are approximately Gaussian. The scatterplot shows the observed values of the informative variables for one sampled dataset of 200 observations. The contour plot shows the theoretical density function of the bivariate distribution for the parameters $\boldsymbol{\Theta}_A$ and $\boldsymbol{\Theta}_B$ chosen to have some probability of overlapping in the tails.

Scenario 2: Low Degrees of Freedom. We assume that the density function of all variables is leptokurtic with fatter tails than a Gaussian. We generate $n = 200$ observations with only two informative variables, $m = 2$, sampled from the mixture of component A with parameters $\boldsymbol{\Theta}_A = \{\boldsymbol{\mu}_A = -3, \boldsymbol{\sigma}_A = 2, \nu_A = 10\}$ and component B with parameter $\boldsymbol{\Theta}_B = \{\boldsymbol{\mu}_B = 4, \boldsymbol{\sigma}_B = 4, \nu_B = 6\}$. The remaining $q = 98$ variables have all been generated from the same distribution. $\boldsymbol{\Theta}_q = \{\boldsymbol{\mu}_q = 0, \boldsymbol{\sigma}_q = 1, \nu_q = 4\}$. In Figure 3.10 we can see from the plot of the two densities that they are more peaked and concentrated around their center but with more extreme realisations.

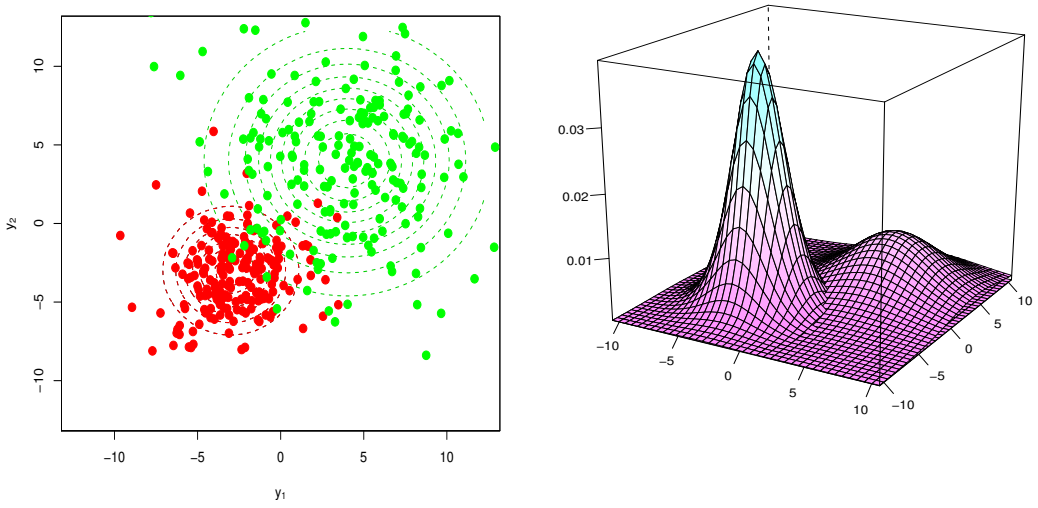

Figure 3.10: Scenario 2: Low degrees of freedom. The bulk of the sampled points in the scatterplot are more concentrated with some outliers.

Scenario 3: Only non-informative variables have low degrees of freedom. We assume that the two informative variables, $m = 2$, are sampled from a mixture of t distributions with high degrees of freedom. Whereas the non informative variables are significantly more $q = 200$ and are sampled from one single heavy tailed t distribution. In Figure 3.11 we can see that the two clusters are normally distributed around their center given our choice of parameters vectors $\boldsymbol{\Theta}_A = \{\boldsymbol{\mu}_A =$

-3, $\boldsymbol{\sigma}_A = 2$, $\nu_A = 40\}$ and $\boldsymbol{\Theta}_B = \{\boldsymbol{\mu}_B = 4$, $\boldsymbol{\sigma}_B = 4$, $\nu_B = 40\}$. The density of the $q = 200$ noise variables instead has low degrees of freedom $\boldsymbol{\Theta}_q = \{\boldsymbol{\mu}_q = 0$, $\boldsymbol{\sigma}_q = 1$, $\nu_q = 4\}$.

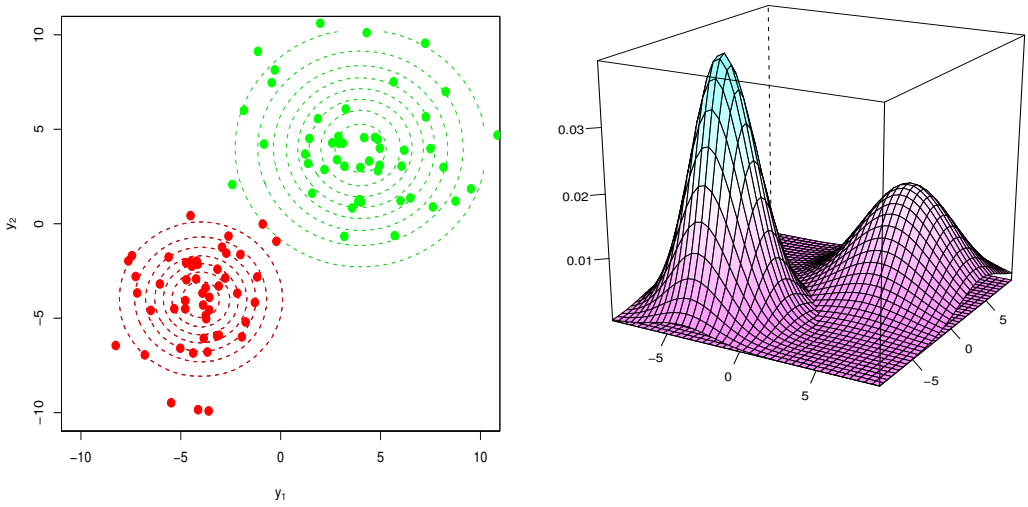

Figure 3.11: Scenario 3: In this scenario we have only a limited number of observations, $n = 100$, while the non informative variable are $q = 200$. As can be seen from the plots the two components are fairly well separated and have high degrees of freedom which make them approximately normal. The non informative variables, not plotted here, have instead very long tails.

Scenario 4: Limited number of high dimensional observations. We generate a more realistic data sample where the number of noise variables is very high $q = 2000$ with only a small fraction of informative variables $m = 20$, a ratio of 1 to 100. We sample only few observations $n = 100$ to stress the ability of the algorithms to reconstruct the original clusters with limited information. The mixing components and the noise generating distribution are all multivariate t Student with fat tails. For component A the parameter vector is $\boldsymbol{\Theta}_A = \{\boldsymbol{\mu}_A = -2$, $\boldsymbol{\sigma}_A = 3$, $\nu_A = 10\}$, for component B is $\boldsymbol{\Theta}_B = \{\boldsymbol{\mu}_B = 4$, $\boldsymbol{\sigma}_B = 2$, $\nu_B = 12\}$ and for non informative variables is $\boldsymbol{\Theta}_q = \{\boldsymbol{\mu}_q = 0$, $\boldsymbol{\sigma}_q = 1$, $\nu_q = 5\}$.

Simulation Results

For each one of the scenarios describe above, we generated over 100 randomly sampled datasets. On each dataset we iterate the EM algorithm 20 times using different sets of initial parameters $\mathbf{\Psi}^0$. The results of all the simulations are summarized in Tables 3.5, 3.6, 3.7 and 3.8 for scenario 1, 2, 3 and 4 respectively. Figure 3.12, 3.13 and 3.14 refer to the first three scenario and visualize the clustering performance over one single sample.

Table 3.5 refers to scenario 1 where mixture components and noise variables both have approximate normal density function, i.e. high degrees of freedom. In this situation we expect Gaussian and t mixture models to perform similarly. This is in fact the case as we can see from the score in column `ARI`. Equivalent penalization models have comparable results with a slight advantage for t mixtures which can be explained with the robustness of this family of distributions.

From the analysis of the results it emerges that the variable selection significantly improves the clustering performance and that the joint penalization of μ and σ achieves the best results. Both $\texttt{PGM}_{\mu,\sigma}$ and $\texttt{PTM}_{\mu,\sigma}$ penalise exactly all the noise variables without any error as `Sens` and `Spec` are 100 at the same time. The overlapping probability of the two components, as was evident from the scatterplot in Figure 3.9, is the only limit to the perfect assignment of all the observations and is within expectations.

Figure 3.12 shows one sampled dataset under scenario 1 where the performance of $\texttt{PGM}_{\mu,\sigma}$ and $\texttt{PTM}_{\mu,\sigma}$ is expected to be very similar. By construction only the two first variables are useful for clustering, y_1 and y_2, and they are correctly selected by both models. The colour coding is proportional to the posterior or responsibility of each component and shows how the uncertainty is bigger at the border where the two densities mix. The accuracy is quite high with no real differences between the two models. The inferred parameters $\mathbf{\Theta}$ are also quite close to the true values of the original density function. The t mixture in particular estimates correctly also the degrees of freedom.

Table 3.6 refers to scenario 2 where both mixture components and noise vari-

Model	ARI	Sens	Spec
GM	3 (3)	100 (0)	0 (0)
$\texttt{PGM}_{\mu}$	1 (1)	0 (0)	100 (0)
$\texttt{PGM}_{\sigma}$	17 (13)	100 (0)	29 (35)
$\texttt{PGM}_{\mu,\sigma}$	71 (13)	100 (0)	100 (0)
TM	7 (9)	100 (0)	0 (0)
$\texttt{PTM}_{\mu}$	19 (9)	100 (0)	25 (22)
$\texttt{PTM}_{\sigma}$	19 (10)	99 (1)	28 (44)
$\texttt{PTM}_{\mu,\sigma}$	72 (13)	100 (0)	100 (0)

Table 3.5: Scenario 1, Summary results of different mixture models when informative and noise variables have approximatively normal density function. Top and bottom half report the performance of different Gaussian and Student's t mixtures respectively: with no penalisation GM and TM, with μ penalisation $\texttt{PGM}_{\mu}, \texttt{PTM}_{\mu}$, with σ penalisation $\texttt{PGM}_{\sigma}, \texttt{PTM}_{\sigma}$ and joint μ and σ penalisation $\texttt{PGM}_{\mu,\sigma}, \texttt{PTM}_{\mu,\sigma}$. For each model we report the average score over the 100 random samples and in brackets the standard deviation of the results.

ables follow a t density function with low degrees of freedom. In this case the Gaussian mixture models underperform t models because they can not properly select the informative variables. $\texttt{PGM}_{\mu}$ penalises all the variables including those that are informative, $\texttt{Sens} = 0$. Instead $\texttt{PGM}_{\sigma}$ does not penalises enough the parameter σ of the informative variables and tries to fit the model using also the noise variables $\texttt{Spec} = 33$.

The t mixture models, instead, give fairly good results in all simulations presented here, with the penalised models showing an higher clustering performance. In this scenario, the regularization of the mean seems to be fairly effective even by itself since $\texttt{PTM}_{\mu}$ correctly identifies all the noise variable, with only few false negatives.

Figure 3.13 refers to scenario 2 where we expect $\texttt{PTM}_{\mu,\sigma}$ to overperform $\texttt{PGM}_{\mu,\sigma}$. The presence of t distributed noise variables induce $\texttt{PGM}_{\mu,\sigma}$ to cluster observations using non-informative variables. By attempting to fit the noise variables it can not identify the original components nor estimate their true parameter values $\boldsymbol{\Theta}$. $\texttt{PTM}_{\mu,\sigma}$ instead, having selected the informative variables, y_1 and y_2, can infer quite accurately the true density function of each component and therefore achieves an

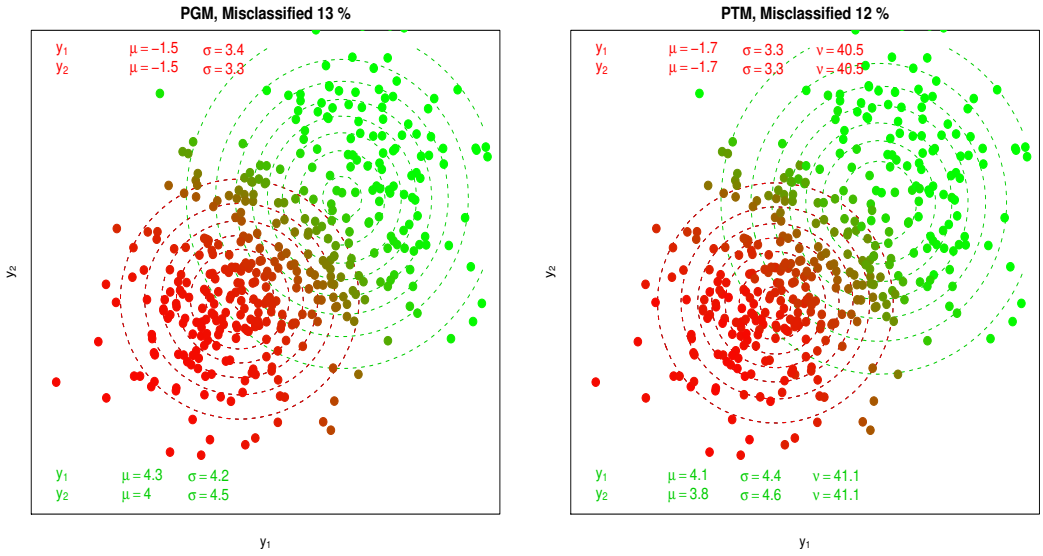

Figure 3.12: Scenario 1. Visualisation of cluster assignment performance of the penalised mixture of Gaussian, `PGM`, and the penalised mixture of t, `PTM` when data are sampled from a mixture of distributions with high degrees of freedom. Colour coding is proportional to the posterior probability τ that the observation has been generated by cluster A, in red, or cluster B in green. In the legend we report the parameter $\boldsymbol{\Theta}_A$ and $\boldsymbol{\Theta}_B$ inferred by the EM algorithm. Note how out of 100 variables both models correctly select the only 2 informative: y_1 and y_2. The accuracy of the two tested models is fairly similar.

higher percentage of correct cluster assignment.

Table 3.7 and Figure 3.14 refer to scenario 3 where the informative variables have been sampled from t distributions with high degrees of freedom which means that Gaussian mixture models should fit them quite accurately. This does not happen because the non-informative variables have long tails and can only be fitted with some degree of precision by the t mixtures models. In Figure 3.14 in particular, we can see that even if the two clusters are fairly well separated and approximately Gaussian, only $\texttt{PTM}_{\mu,\sigma}$ correctly selects the two informative variables y_1 and y_2 and assign almost all the observations to the right cluster. $\texttt{PGM}_{\mu,\sigma}$ instead can not filter out the noise which ends up masking the true clusters.

Table 3.8 refers to scenario 4 where we only have few high dimensional t dis-

Model	ARI	Sens	Spec
GM	2 (1)	100 (0)	0 (0)
PGM_{μ}	2 (1)	0 (0)	100 (0)
PGM_{σ}	2 (1)	100 (0)	33 (18)
$\text{PGM}_{\mu,\sigma}$	2 (1)	100 (0)	46 (43)
TM	62 (13)	100 (0)	0 (0)
PTM_{μ}	74 (4)	91 (1)	100 (0)
PTM_{σ}	67 (13)	83 (9)	95 (8)
$\text{PTM}_{\mu,\sigma}$	85 (3)	96 (7)	100 (0)

Table 3.6: Scenario 2, Summary results of tested mixture models when informative and noise variables have t density function with low degrees of freedom. Note how the clustering performance of t mixture models is significantly better and improves with penalisation, column ARI. Similarly the variable selection is fairly accurate as it show very low false nagatives and false positive errors.

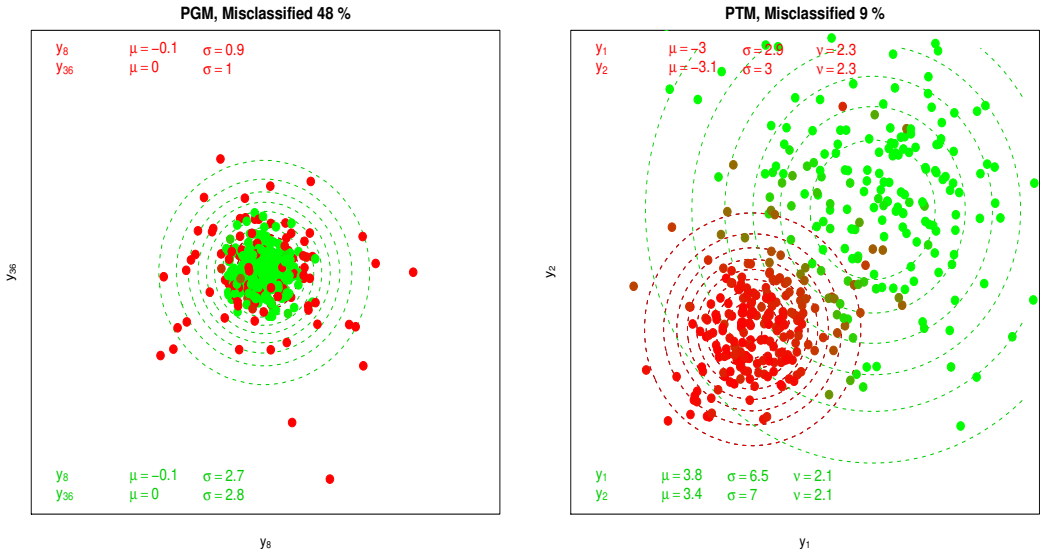

Figure 3.13: Scenario 2: Informative and non informative variables have been sampled from t density function with low degrees of freedom. PGM fails to identify the only informative variables and tries to cluster the noise variables.

tributed observations and the number of informative variables is only a small fraction of the total. Robust penalisation is essential to filter out the 2000 noise

Model	ARI	Sens	Spec
GM	3 (3)	100 (0)	0 (0)
$\texttt{PGM}_{\mu}$	6 (3)	11 (23)	34 (19)
$\texttt{PGM}_{\sigma}$	6 (3)	82 (34)	47 (45)
$\texttt{PGM}_{\mu,\sigma}$	8 (6)	77 (39)	40 (45)
TM	8 (8)	100 (0)	0 (0)
$\texttt{PTM}_{\mu}$	48 (14)	76 (23)	56 (50)
$\texttt{PTM}_{\sigma}$	53 (8)	65 (15)	48 (46)
$\texttt{PTM}_{\mu,\sigma}$	87 (19)	100 (0)	92 (7)

Table 3.7: Scenario 3, Summary results when we have few observations with 200 non informative variables and only two informative. Even if the density of the mixture components is approximately Gaussian, the performance of the Gaussian models is pooor because they can not filter out the long tailed noise variables. Penalised t mixtures, on the ther hand, are robust to noise data and, by selecting only the relevant variables, can accurately identify the true clusters.

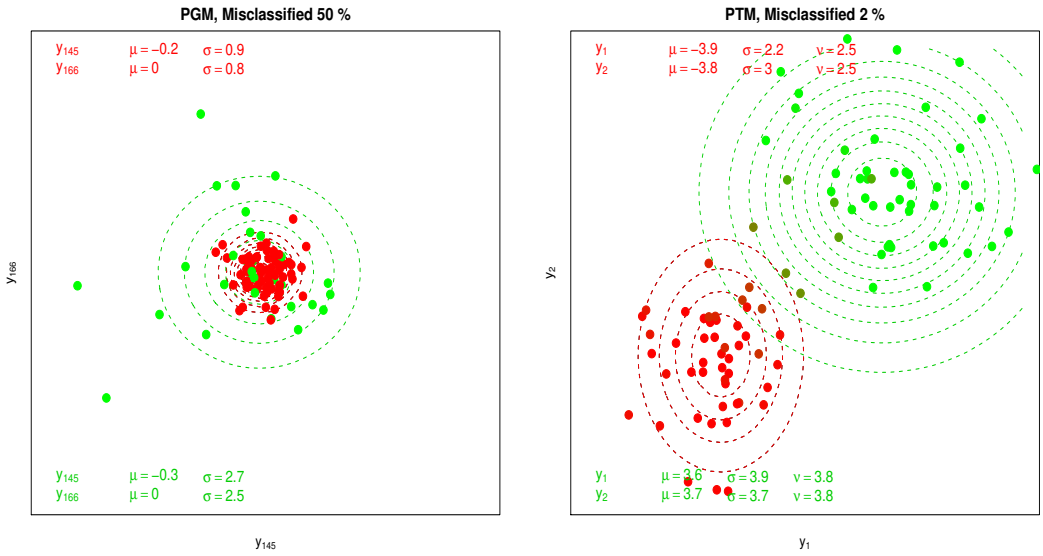

Figure 3.14: Scenario 3: We have a limited number of observations with only 2 informative variables and 200 non informative variables. Mixture components are approximately Gaussian and fairly well separated but only $\texttt{PTM}_{\mu,\sigma}$ can accurately exclude all long tailed non informative variables. Being able to identify the only two informative variables, y_1 and y_2 can reconstruct the true cluster.

variables and select the only 20 that can help us identify the true clusters. In fact the performance of the Gaussian mixture models `PGM` is poor because it either penalises all the variables, excluding also those informative, or it does not exclude any variable at all, failing to unmask the hidden structure of the data. On the other hand, as we can see from the column `ARI`, the accuracy of `PTM` increases considerably when it jointly regularizes μ and σ because it is able to filter out almost all the noise variables.

Model	`ARI`	`Sens`	`Spec`
`GM`	2 (3)	100 (0)	0 (0)
$\texttt{PGM}_{\mu}$	1 (3)	13 (4)	34 (10)
$\texttt{PGM}_{\sigma}$	1 (2)	70 (9)	14 (35)
$\texttt{PGM}_{\mu,\sigma}$	4 (5)	100 (1)	15 (30)
`TM`	34 (14)	100 (0)	0 (0)
$\texttt{PTM}_{\mu}$	48 (17)	53 (24)	85 (43)
$\texttt{PTM}_{\sigma}$	55 (16)	58 (22)	76 (41)
$\texttt{PTM}_{\mu,\sigma}$	79 (22)	92 (23)	88 (20)

Table 3.8: Scenario 4, a more realistic case where observed data is high dimensional, $p = 2020$, with only few observations, $n = 100$, and the ratio between informative and non-informative variables is 1 to 100. Penalised mixture of t are robust to long-tailed noise data and by selecting only the relevant dimensions can improve the clustering performance.

3.5.6 High Dimensional Settings

In this section we assume more difficult conditions and simulate scenarios which are much closer to the real life problems we want to investigate, like microarray data.

We assume a sample size of $n = 200$, and $m = 20$ variables informative for clustering. The number of uninformative variables, q, is always taken to be much higher than m. As always, the informative variables are sampled from a mixture of K multivariate Student's t distributions. All the uninformative variables share the same parameters and also follow a Student's t distribution.

In order to explore the effects of having fatter tails on both clustering and variable selection performance, we consider two scenarios: a low degrees of freedom case (Low DoF), in which the distributions have tails that are more pronounced compared to multivariate Gaussians, and a high degrees of freedom case (High DoF), that is Gaussian distributions. We simulate data with both two and tree components. When $K = 2$, the parameters of the densities are chosen to ensure that there is roughly a 30% overlap between them; when $K = 3$, which we indicate as A, B and C, the parameters are chosen so that there is about 25% overlap between A and B, 30% overlap between A and C, and around 5% overlap between B and C. The exact density parameters of the two components for $K = 2$ are $\boldsymbol{\Psi}_A = \{\pi = 0.5, \mu = -2, \sigma = 2, \nu = 7\}$ and $\boldsymbol{\Psi}_B = \{0.5, 2, 5, 7\}$ respectively; for $K = 3$ are $\boldsymbol{\Psi}_A = \{0.3, -6, 4, 7\}$, $\boldsymbol{\Psi}_B = \{0.2, 0, 8, 7\}$ and $\boldsymbol{\Psi}_C = \{0.5, 6, 12, 7\}$. In the high Degrees of Freedom scenario all ν are set to 30.

In a separate experiment we specifically assess the effects of increasing the number of uninformative variables, and consider two cases: $q = 200$ (low noise) and $q = 2000$ (high noise), while still keeping the number of informative variables fixed at $m = 20$.

To assess the clustering performance of the suggested penalised Student's t mixture model (`PTM`), we compare it with two competing clustering methods that simultaneously partition the samples and identify the relevant genes: the penalised Gaussian mixture (`PGM`) (Xie et al., 2008a), and the sparse K-means algorithm (`PKM`) (Witten and Tibshirani, 2010), which uses a lasso-type penalty to select the variables and obtain a sparse hierarchical clustering.

For each setting being considered, we generate 100 independent data sets and report on Monte Carlo averages. In Table 3.9 we consider two and three clusters, and fit the three competing models. In all cases, the correct number of clusters is pre-specified and only the number of selected variables is learnt from the data using model selection. The modified BIC criterion is used to select the degree of penalty in both `PGM` and `PTM` models, without using the resampling procedure, which is evaluated separately later. For `PKM`, we use the built-in model selection procedures that rely on multiple permutations of the data. All methods are assessed using

ARI, sensitivity and specificity indexes.

In the low degrees of freedom and $K = 2$ case, PTM achieves the highest sensitivity index at the cost of a marginally lower specificity compared to the other two models. This ability to identify and retain the truly informative variables translates into the highest average ARI for PTM. As expected, PKM performs poorly in this case as no probabilistic model is assumed, and the model is more sensitive to extreme observations.

In the $K = 3$ case, the specificity of all three competing models is still comparable, but PTM achieves the highest sensitivity that leads to the best clustering performance. When the distributions are Gaussian, both PGM and PGM have similar performances, as expected in this case, whereas performance of PKM is lower, especially with three clusters.

		$K = 2$			$K = 3$		
DoF	Model	ARI	Sens	Spec	ARI	Sens	Spec
Low	PKM	0.33	0.38	1.00	0.10	0.20	1.00
	PGM	0.57	0.60	0.96	0.40	0.70	0.98
	PTM	0.72	0.76	0.97	0.56	1.00	0.96
High	PKM	0.62	0.66	1.00	0.37	0.77	1.00
	PGM	1.00	1.00	1.00	0.71	1.00	0.99
	PTM	1.00	1.00	1.00	0.71	1.00	0.99

Table 3.9: Performance assessment of three competing sparse clustering methods. Data simulated with parameters $n = 200$, $m = 20$, and $q = 2000$. The correct number of mixture components is assumed known, and variable selection is performed for each simulated data set.

Resampling Strategies

We verify that even in more extreme conditions with high dimensional dataset, the contribution of the resampling procedure is not only useful for variable ranking, but it is also critical to improve model selection accuracy.

The potential benefit that can be gained from the resampling procedure of Section 3.4.1, in terms of both variable selection and clustering performance, are

summarised in Table 3.10. Here we consider four scenarios whereby we vary the distribution used to generate the data within each cluster as well as the number of uninformative variables. The data are sampled from a mixture of three multivariate Student's t distributions with parameters: $\boldsymbol{\Psi}_A = \{\pi = 0.3, \mu = -6, \sigma = 4, \nu = 10\}$, $\boldsymbol{\Psi}_B = \{0.3, 0, 8, 10\}$ and $\boldsymbol{\Psi}_C = \{0.4, 6, 16, 10\}$. The sample size is $n = 200$, with $m = 20$ informative variables while the number of uninformative variables q is taken to be both 200 and 2000. After running the subsampling procedure with a fixed probability selection threshold, $\tilde{\pi} = 0.7$, the improved ability of the model to exclude the noise variables improves the clustering performance, as quantified by the `ARI`. The improvement is particularly remarkable when the distributions have longer tails.

	Low DoF		High DoF	
Resampling	$q = 200$	$q = 2000$	$q = 200$	$q = 2000$
No	0.84	0.64	0.88	0.82
Yes	0.92	0.86	0.98	0.93

Table 3.10: ARI for the PTM model, with and without resampling. Data simulated with parameters $K = 3, n = 200, m = 20$. In the High DoF scenario all ν are set to 30. PTM was fitted using $K = 3$

	Low DoF		High DoF	
Resampling	$q = 200$	$q = 2000$	$q = 200$	$q = 2000$
No	0.5 (3.7)	0.0 (4.7)	0.72 (3.3)	0.55 (3.75)
Yes	0.85 (2.85)	0.3 (2.35)	1.00 (3)	1.00 (3)

Table 3.11: Percentage of correctly identified mixture components in the PTM model, with and without resampling. Data simulated with parameters $K = 3, n = 200, m = 20$. The average number of clusters is in brackets.

In Table 3.11 we explore the effects of the two-step resampling approach described in Section 3.4.1 on the selection process of the number of mixture components. For this experiment, we use the simulated data sets used to produce the results of Table 3.10, where the true number of clusters is $K = 3$. For each scenario being considered, we search among models having up to five clusters. We report

on the percentage of times the correct number of clusters is selected by the two strategies, with and without resampling. As expected, both model selection strategies perform better when the distribution of the simulated variables is Gaussian. In both high and low degrees of freedom scenarios, there is a notable performance gain when using the resampling scheme especially so in particularly demanding conditions when the data have fatter tails and the number of noise variables is high. We note also that in all scenarios the average number K selected, quoted in brackets in Table 3.11, is higher for the first strategy. This evidence confirms that, by reducing the number of noise variables, the resampling approach alleviates the overfitting problem which is particularly important in high dimensional setting.

3.6 Discussion

When applied to high dimensional datasets, many clustering techniques begin to suffer from the curse of dimensionality, degrading the quality of the results. We have developed a method for simultaneously clustering high dimensional data and selecting informative variables by employing a penalised mixture of Student's t distributions. We chose finite mixture of t in the first place because it is robust to outliers and noisy observations, but we impose also penalisation to improve its clustering performance through variable selection. We also described a resampling procedure to support the model selection and provide an effective metric to rank variables.

We have proposed an adaptive L_1-norm penalty function to regularize the maximum likelihood estimates of the location μ and the dispersion parameter σ in order to identify the non informative variables. We have derived a modified EM algorithm to find the MLE of all the unknown parameters of the mixture of distributions and then discussed the appropriate likelihood based criteria to select the best model.

We have then tested the proposed clustering algorithm on different simulated scenarios and given evidence that, as the tails of the distribution of the informative and noise variables becomes longer, penalised mixture of Student's t distributions

achieves higher clustering accuracy than the equivalent Gaussian model. We have also verified the benefit of implementing a subsampling routine to refine the variable selection process and increase the model selection accuracy.

While the results so far have responded to our expectations, we feel there is still scope to further extend the research in this area. On the same path of the present study, we believe that investigating other types of penalty functions could lead to some interesting results. For example, in the context of Gaussian mixtures, Xie et al. (2008a) proposed to group together multiple parameters of the same variable across clusters with the idea of performing a more effective model regularization. Similarly Wang and Zhu (2008) suggested to use any prior knowledge available to group together variables that are perceived to be either all informative or all non-informative. A slightly different approach has been described by Liu and Rattray (2010) who proposed pairwise variable selection that retains only those variables that can help to separate at least two clusters. All of these approaches have been tested on t mixture and, as we have seen before, combining robust modelling and regularization could lead to interesting results.

Another possible route worth following is to impose regularization on the degrees of freedom parameter. We envisage that ν should be part of the set of relevant parameters considered when performing variable selection. We can imagine the situation where two clusters share the same location and show the same dispersion but might be characterized by different tail shape. The challenge, in this case, is to recognise exactly whether the observed data are sampled from a long tailed t distribution or rather from a mixture of two or more Gaussian distribution. This problem would necessarily require the development of alternative ways of combining penalties on different parameters.

So far, we have accepted the restrictive assumption that the correlation matrix of recorded variables was diagonal, i.e. independent variables. This simplifying hypothesis was formulated to make the inversion of the covariance matrix computationally cheaper by avoiding singularities. Relaxing this assumption could lead to improved results, especially in the situation where two or more variables show some correlation. A similar event would signal the importance of those variables

even if their parameters' estimate deviate only marginally from the population average.

Having tested the performance of the penalised mixture of t distributions on simulated datasets, in the following chapters we will use the proposed model to investigate the two real life problems we described in chapter 1.

In order to solve the bioinformatics problem, in chapter 5 we will fit the mixture model on the microarray dataset to identify possible cancer subtypes and select only those genes that are really informative. One reason for using t mixture models is that gene expression levels have been observed to have distributions with heavier tails than the Gaussian. The other reason is that the proposed penalisation can reduce significantly the number of genes necessary to identify different subtypes and can also help to discover which genes, out of the few thousands recorded, are responsible for the insurgency of specific tumor subtypes.

The second application we discuss in chapter 6 is in finance. It is standard practice to group financial markets in macro sectors based on the nature and fundamental characteristics of the goods exchanged. We suggest, instead, a data-driven approach and cluster financial markets on the basis of the observed features of each market's price dynamics. As we do not have any prior knowledge of which feature are more informative to cluster markets, we let the penalised mixture of t distribution answer that question for us. The results should hopefully lead us to engineer more robust trading strategies that are better suited to exploit the specific features of each cluster.

3.A Appendix: Derivations

3.A.1 Conditional Expectation $Q_{2,k}$ and $Q_{3,k}$

$$Q_{2,k}(\boldsymbol{\nu}_k|\boldsymbol{\Psi}^{(t-1)}) = -\log\Gamma(\frac{\nu_k}{2}) + \frac{\nu_k}{2}\log\Gamma(\frac{\nu_k}{2}) + \frac{\nu_k}{2}\left\{\log u_{i,k}^{(t)} - u_{i,k}^{(t)} + \psi\left(\frac{\nu_k^{(t-1)}+p}{2}\right) - \log\left(\frac{\nu_k^{(t-1)}+p}{2}\right)\right\} - \log u_{i,k}^{(t)} - \psi\left(\frac{\nu_k^{(t)}+p}{2}\right) + \log\left(\frac{\nu_k^{(t)}+p}{2}\right) \tag{3.18}$$

$$Q_{3,k}(\boldsymbol{\mu}_k,\, \boldsymbol{\Sigma}_k|\boldsymbol{\Psi}^{(t-1)}) = \{ -\frac{p}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| + \frac{p}{2}\log u_{i,k}^{(t-1)} - \frac{1}{2}u_{i,k}(\boldsymbol{y}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{y}_i - \boldsymbol{\mu}_k)\} \tag{3.19}$$

3.A.2 First Derivative of $Q_2(\boldsymbol{\Psi})$

$$\frac{\partial}{\partial\nu_k}\left\{\sum_{i=1}^{n}\tau_{i,k}^{(t)}\, Q_{2,k}(\boldsymbol{\nu}_k|\boldsymbol{\Psi}^{(k-1)})\right\} = -\psi(\frac{\nu_i}{2}) + \log(\frac{\nu_i}{2}) + 1 + \frac{\sum_{j=1}^{n}\tau_{i,j}^{(k)}(\log u_{ij}^{(k)} - u_{ij}^{(k)})}{\sum_{j=1}^{n}\hat{\tau}_{i,j}^{(k)}} + \psi\left(\frac{\nu_i+p}{2}\right) - \log\left(\frac{\nu_i+p}{2}\right) = 0 \tag{3.20}$$

3.A.3 Updating algorithm for $\boldsymbol{\mu}$

Since the conditional expectation of the complete data $Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})$ defined in (3.7) is concave-differentiable with respect to $\mu_{k,d}$, when $\mu_{k,d} \neq 0$ we know a local maximum $\mu_{k,d}$ must satisfy the following conditions:

$$\begin{cases} \frac{\partial}{\partial\mu_{k,d}}Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)}) = 0 & \text{if and only if} \quad \mu_{k,d} \neq 0 \\ Q_p(0,.) \geq Q_p(\Delta\mu_{k,d},.) & \text{if and only if} \quad \mu_{k,d} = 0 \end{cases}$$

Case I: $\mu_{k,d} \neq 0$

$$\frac{\partial}{\partial\mu_{k,d}}\left\{\sum_{i=1}^{n}\tau_{i,k}^{(t)}\left(-\frac{1}{2}\frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(y_{i,d} - \mu_{k,d})^2\right) - \lambda_\mu\,\omega_{k,d}\,|\mu_{k,d}|\right\} = 0$$

which yields

$$\sum_{i=1}^{n} \tau_{i,k}^{(t)} \frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(y_{i,d} - \mu_{k,d})^2 = \lambda_\mu \, \omega_{k,d} \, |\mu_{k,d}|$$

Case II: If $\mu_{i,d} = 0$ is a maximum then we compare $Q_p(0, .)$ with $Q_p(\Delta\mu_{k,d}, .)$.

$$\begin{aligned}
& Q_p(\mathbf{0}, .) \geq Q_p(\Delta\mu_{k,d}, .) \qquad \text{for any} \quad \Delta\mu_{k,d} \quad \text{near} \quad 0 \\
\Longrightarrow & -\textstyle\sum_{i=1}^{n} \frac{\tau_{i,k}^{(t)}}{2} \frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(y_{i,d})^2 + C_1 \geq -\sum_{i=1}^{n} \frac{\tau_{i,k}^{(t)}}{2} \frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(y_{i,d} - \Delta\mu_{k,d})^2 - \lambda_\mu \, \omega_{k,d} \, |\Delta\mu_{k,d}| + C_1 \\
\Longrightarrow & \textstyle\sum_{i=1}^{n} \frac{\tau_{i,k}^{(t)}}{2} \frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(2 \, y_{i,d} \, \mathrm{sign}(\Delta\mu_{k,d}) - |\Delta\mu_{k,d}|) \leq \lambda_\mu \, \omega_{k,d} \\
\Longrightarrow & \frac{\sum_{i=1}^{n} |\tau_{i,k}^{(t)} \, u_{i,k}^{(t)} y_{i,d}|}{\sigma_{k,d}^2} \leq \lambda_\mu \omega_{k,d}
\end{aligned}$$

3.A.4 Updating algorithm for $\boldsymbol{\sigma}$

Since the conditional expectation of the complete data $Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})$ defined in (3.7) is concave-differentiable with respect to $\sigma_{k,d}^2$ when $\sigma_{k,d}^2 \neq 1$ we know a local maximum $\sigma_{k,d}^2$ must satisfy the following conditions:

$$\begin{cases} \frac{\partial}{\partial \sigma_{k,d}} Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)}) = \mathbf{0} & \text{if} \quad \sigma_{k,d} \neq 1 \\ Q_p(\mathbf{0}, .) \geq Q_p(\Delta\sigma_{k,d}^2, .) & \text{if} \quad \sigma_{k,d}^2 = 1 \quad \text{for any} \quad \Delta\sigma_{k,d}^2 \quad \text{near} \quad 0 \end{cases} \tag{3.21}$$

Notice that

$$Q_p(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)}) = C_1 + \sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2} \log \sigma_{k,d}^2 + C_2 - \frac{1}{2} \frac{u_{i,k}^{(t)}}{\sigma_{k,d}^2}(y_{i,d} - \mu_{k,d})^2 \right) - \lambda_\sigma \, \omega_{k,d} \, |\log \sigma_{k,d}^2| + C_3$$

where C_1, C_2 and C_3 are constants with respect to $\sigma_{k,d}^2$. Therefore the first equation of (3.21) becomes

$$\sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2 \, \hat{\sigma}_{i,k}^2} + \frac{u_{i,k}^{(t)}(y_{i,d} - \mu_{k,d})^2}{2 \, \hat{\sigma}_{k,d}^4} \right) - \frac{\lambda_\sigma \, \omega_{k,d} \, \mathrm{sign}\left(\log \hat{\sigma}_{k,d}^2\right)}{\hat{\sigma}_{k,d}^2} = 0 \qquad \text{if} \quad \hat{\sigma}_{k,d} \neq 1$$

The second equation of (3.21) becomes

$$LHS = C_1 + \sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2}\log(1) \; + C_2 - \frac{u_{i,k}^{(t)}}{2} \frac{(y_{i,d} - \mu_{k,d})^2}{1} \right) - \lambda_\sigma \, \omega_{k,d} \, |\log(1)| + C_3$$

$$RHS = C_1 + \sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2}\log(1 + \Delta\sigma_{k,d}^2) \; + C_2 - \frac{u_{i,k}^{(t)}}{2} \frac{(y_{i,d} - \mu_{k,d})^2}{(1 + \Delta\sigma_{k,d}^2)} \right) - \lambda_\sigma \, \omega_{k,d} \, |\log(1 + \Delta\sigma_{k,d}^2)| + C_3$$

and thus

$$\frac{1}{2} \sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(\log(1 + \Delta\sigma_{k,d}^2) \; - u_{i,k}^{(t)} \frac{(y_{i,d} - \mu_{k,d})^2}{(1/(1 + \Delta\sigma_{k,d}^2) \; - \; 1)} \right) \leq \lambda_\sigma \, \omega_{k,d} \, |\log(1 + \Delta\sigma_{k,d}^2)|$$

Using Taylor's expansion we have

$$\sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2} \; + \frac{u_{i,k}^{(t)} \, (y_{i,d} - \mu_{k,d})^2}{2} \right) \Delta\sigma_{k,d}^2 + O((\Delta\sigma_{k,d}^2)^2) \leq \lambda_\sigma \, \omega_{i,d} \, |\log(1 + \Delta\sigma_{k,d}^2)|,$$

leading to

$$\sum_{i=1}^{n} \tau_{i,k}^{(t)} \left(-\frac{1}{2} \; + \frac{u_{i,k}^{(t)} \, (y_{k,d} - \mu_{k,d})^2}{2} \right) \text{sign}(\Delta\sigma_{i,d}^2) + O(|\Delta\sigma_{i,d}^2|) \leq \lambda_\sigma \, \omega_{k,d} \left| \frac{\log(1 + \Delta\sigma_{k,d}^2)}{\Delta\sigma_{k,d}^2} \right|.$$

Chapter 4

Mixture of Lasso Regressions with t-Errors

4.1 Introduction

Our interest in the previous chapter was to perform unsupervised cluster analysis with variable selection. Here, we consider a slightly different situation where paired explanatory and response variables are available and a supervised learning approach would be more appropriate.

We noted before that in the real life data under investigation, some of the response variables might be explained by only a small subset of the associated input variables. Maintaining the assumption that the recorded samples have been generated from an heterogeneous population, we propose here a mixture of regressions model that simultaneously regularizes the regression coefficients and selects the relevant covariates. More precisely, we follow a Bayesian approach which provides the most suitable framework and define a hierarchical structure of priors that allows us to build a model with the desired properties of accuracy and parsimony.

Since the final model is necessarily complex and high dimensional, we devise an efficient sampling algorithm based on Particle Markov Chain Monte Carlo methods (Andrieu et al., 2010) to support the practical implementation of the estimation procedure. In addition to giving an account of the simulation procedure we also test

its performance using experimental data and illustrate the results that demonstrate the accuracy of the model.

It should be noted that, while mixture of regressions models can be applied to a wider range of inferential problems, here we focus our attention on its clustering and variable selection capabilities and our following discussion reflects that.

The outline of the chapter is as follows. In section 4.2 we describe the hierarchical representation of the model and justify the choice of priors that lead to the posteriors of interest. In section 4.3 we discuss how we can meet some of the estimation challenges posed by the model and how we can efficiently simulate from the target posterior distribution using a combination of Sequential Monte Carlo sampling algorithms and metropolised Gibbs sampling steps. In section 4.4 we use a variety of numerical examples to illustrate the simulation strategy and give evidence of the main properties of the proposed model. In appendix 4.A we review the essential elements of Markov Chain Monte Carlo methods and the Gibbs sampling algorithm in particular.

4.2 The Model

Generalising the peculiarities of the financial and microarray data we want to investigate, let us first highlight the relevant aspects of the problem that motivate the mixture of regression model we propose.

Assume we have a collection of $n \in \mathbb{N}^+$ paired observations $\mathcal{D}_n = (\boldsymbol{x}_i, y_i)_{i=1}^n$ where $y_i \in \mathbb{R}$ is the response variable and $\boldsymbol{x}_i \in \mathbb{R}^p$ is the corresponding vector of explanatory variables. To simplify notation we use $x_{1:p,i}$ to indicate the collection of covariates at the i^{th} sample and set the first element $x_{1,i}$ to be 1 to allow a more convenient formulation of the model. The defining characteristic of the data is that the n samples are independently generated from an heterogeneous population and only few of the p covariates convey any useful information to explain y_i.

To answer these demanding conditions, we propose a Bayesian mixture model which postulates that there are $K \leq n$ possible linear regression curves to describe the data and that each curve potentially depends upon a different collection of the

variables $1, \ldots, p$. To facilitate the derivation of a sparse solution, we introduce a p-dimensional binary vector $\gamma_{1:p}^k$, where we use $\gamma_{1:p}^k$ to denote $(\gamma_1^k, \ldots, \gamma_p^k)$, which encodes whether each of p observed covariates should be included or not in the k^{th} regression curve for $k = 1, \ldots, K$. Similarly, we use $\gamma_{1:p}^k$ as a subscript indicator which deletes the elements corresponding to $\gamma_d^k = 0$ for $d \in \{1, \ldots, p\}$ and returns a vector of length $|\gamma_{1:p}^k|_1$ ($\mathbb{L}_1-$norm).

In a probabilistic framework, the model is then defined as the conditional distribution of y_i given $\boldsymbol{x}_i$

$$y_i|\boldsymbol{x}_i, \boldsymbol{\beta}, \boldsymbol{w}, \boldsymbol{s}_i, \boldsymbol{\gamma}_{1:p} \sim \sum_{k=1}^{K} w_k \mathcal{N}(x'_{\gamma_{1:p}^k, i} \beta_{\gamma_{1:p}^k}^k, s_i^k) \tag{4.1}$$

which is a mixture of normal distributions with parameters

- w_k with $0 \leq w_k \leq 1$ for $k = \{1, \ldots, K\}$ such that $\sum_{k=1}^K w_k = 1$, are the mixing proportion of the K components.
- $\beta_{1:p}^k$ with $\beta_d^k \in \mathbb{R}$ for $d = \{1, \ldots, p\}$, is the collection of regression coefficients.
- s_i^k, with $s_i^k \in \mathbb{R}^+$ for $i = \{1, \ldots, n\}$ is a variable introduced to allow a Student's t regression error.

Having defined the model, the values of the parameters $\boldsymbol{\Psi} = (\boldsymbol{w}, \boldsymbol{\beta}, \boldsymbol{s}, \boldsymbol{\gamma})$ are unknown and will have to be inferred from the data $\mathcal{D}_n$ using a Bayesian approach.

Note that throughout our discussion we assume that the number of clusters K is known. In a different situation, we could have included K in the set of unknown parameters and modified the estimation process accordingly. While this would be a standard procedure, it adds another level of complexity to the model that we rather avoid here since it is not the focus of our investigation. Nonetheless we will consider this extension of the model as a potential subject of future work.

4.2.1 Prior Specification

Whilst a mixture of Gaussian distributions as described in (4.1) is a fairly general model, it is also flexible enough to allow us to choose convenient priors that achieve

the objective of making the model robust to outliers and selecting only the relevant covariates. This task is facilitated by using a hierarchical representation of the mixture model and having different levels of priors and hyperpriors.

Following the standard missing data approach, see Diebolt and Robert (1994), we introduce, for every $i^{th}-$data point, the latent allocation variable $z_i \in \{1, \ldots, K\}$ which indicates the membership of y_i to the $k^{th}-$cluster. Thus, we can simplify the mixture structure and note that the conditional distribution of y_i given $z_i = k$, with probability $p(z_i = k) = w_k$, is the Gaussian distribution

$$y_i \,|\, \gamma^k_{1:p},\, x^k_{\gamma^k_{1:p}},\, \beta^k_{\gamma^k_{1:p}},\, s^k_i,\, z_i = k \sim \mathcal{N}(x'_{\gamma^k_{1:p},i}\, \beta^k_{\gamma^k_{1:p}}, s^k_i). \tag{4.2}$$

Assuming the mixture weights follow a Dirichlet distribution, the prior on $w_{1:K-1}$ is

$$w_{1:K-1} \sim \mathcal{D}ir(\delta)$$

where $\mathcal{D}ir(\delta)$ is the symmetric Dirichlet distribution. Having only the concentration parameter δ specified means we do not have any prior knowledge favouring one component over another, but we still can control how evenly spread the weights $\boldsymbol{w}$ are.

Distribution of s^k_i

Following the hierarchical representation, given $z_i = k$, the prior distribution of the variance parameter s^k_i in (4.2), is set to be

$$s^k_i \sim \mathcal{G}a(d/2, d/2)$$

where $\mathcal{G}a(a, b)$ is Gamma distribution of mean a/b. The hyperparameter d corresponds to the degrees of freedom of the student-t distribution obtained integrating an infinite mixture of normal over a gamma distributed variance parameter. The choice of a lower degrees of freedom parameter d allow us to build a robust regression model that can accommodate for observations errors or more extreme outliers.

In Figure 4.1 we can see how the shape of the prior changes as the degrees of freedom increase.

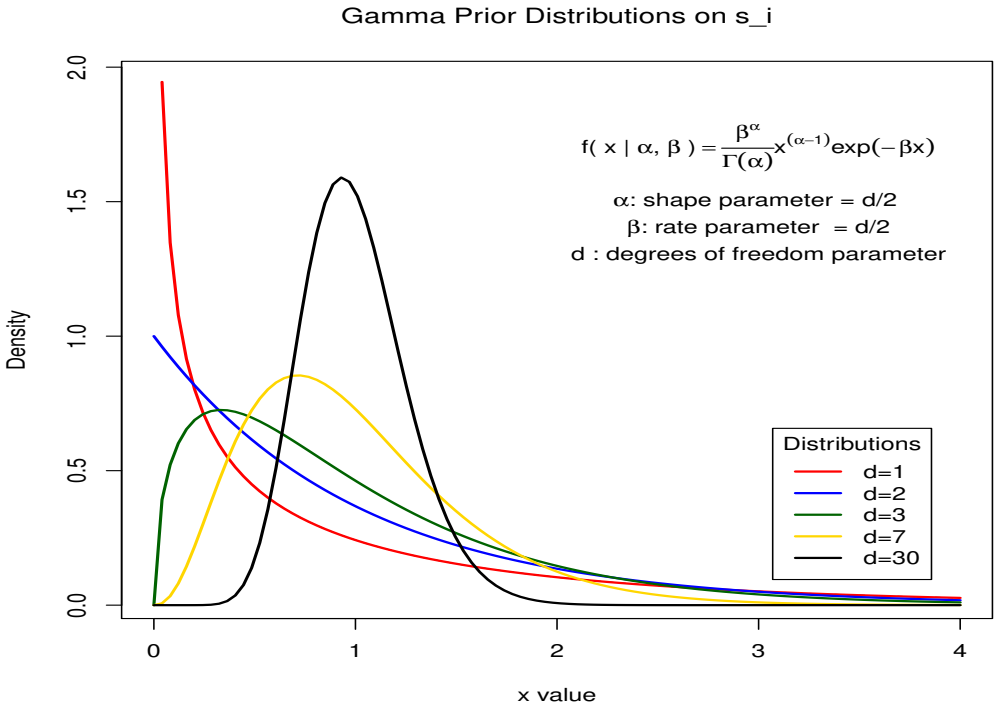

Figure 4.1: Gamma prior distribution on the dispersion parameter s_i^k. We can see how the shape of the distribution changes as a function of the degrees of freedom of the Student's t regression error.

The Bayesian Lasso

A very important feature of the model we propose is that combines, in a mixture framework, shrinkage and variable selection. It achieves this result by adopting specific priors for the regression coefficients $\boldsymbol{\beta}$ and the binary indicator variables $\boldsymbol{\gamma}$.

We have seen in section 2.6 that, using a ML approach in a single mixture component framework, Tibshirani (1996) was able to regularise the estimated linear regression coefficients $\beta_{1:p}^k$ introducing the penalty term: $h_\lambda(\beta_{1:p}^k) = \sum_{d=1}^p |\beta_d^k|^q$ for some $q \geq 0$ and $\lambda_k \in \mathbb{R}^+$. The effect of penalising the likelihood function is to

shrink the vector of MLE of $\beta^k_{1:p}$ toward zero with the possibility of setting some coefficients exactly equal to zero, thus excluding them from the model.

As Park and Casella (2008) point out, equivalent results to the Lasso penalty can be achieved by assuming that $\beta^k_{1:p}$ have independent Laplace, i.e. double-exponential priors,

$$p(\beta^k_{1:p}|\sigma^2_k) = \prod_{d=1}^{p} \frac{\lambda_k}{2\sqrt{\sigma^2_k}} \exp\left(\frac{-\lambda_k|\beta^k_d|}{\sqrt{\sigma^2_k}}\right) \tag{4.3}$$

where $\sigma^2_k \in \mathbb{R}^+$ determines the scaling of the regression coefficients in the $k^{th}-$curve and $\lambda_k \in \mathbb{R}^+$ is the smoothness parameter that controls the tail decay. Since the mass of (4.3) is quite highly concentrated around zero with a distinct peak at zero, the regression coefficient estimates corresponding to the posterior mean and posterior mode are shrunk towards zero in equivalent fashion to the penalisation least squares estimation procedure.

Another convenient property of the double-exponential distribution is that it can be represented as a scale mixture of normals with exponential mixing distribution. Therefore, introducing a latent vector of scale variables we obtain a more tractable hierarchical formulation of the prior on $\beta^k_{1:p}$, see (Hans, 2009). Ignoring for the moment the $\boldsymbol{\gamma}_{1:p}$ indicator and assuming a single component mixture, consider the following hierarchical prior on the d^{th} regression coefficient: $\beta_d|\tau^2_d, \lambda \sim \mathcal{N}(0, \tau^2_d)$ where the hyperparameter τ^2_d itself has hyperprior $\tau^2_d \sim \mathcal{E}x(\lambda^2/2)$. We note that marginally β_d still follows a Laplace distribution with parameter λ

$$\begin{aligned} p(\beta_d) &= \int_0^\infty p(\beta_d|\tau^2_d)\, p(\tau^2_d)\, d\tau^2_d \\ &\propto \exp(-\lambda\,|\beta_d|). \end{aligned}$$

The modular structure of hierarchical modelling allows us to extend, in a straightforward way, the Bayesian Lasso method to our proposed mixture of linear regression. Together with the prior on $\beta^k_{\gamma^k_{1:p}}$ we also specify priors on the hyper-parameters σ_k, with $\sigma_k \in \mathbb{R}^+$, to control the scaling, and $\tau^k_{1:p}$, with $\tau^k_d \in \mathbb{R}^+$, to

induce shrinkage on the coefficients of the k^{th} regression curve.

$$\begin{aligned}
\beta^k_{\gamma^k_{1:p}}|\sigma^2_k, \tau^{2,k}_{\gamma^k_{1:p}}, \gamma^k_{1:p} &\sim \mathcal{N}_{|\gamma^k_{1:p}|_1}\left(0, \sigma^2_k \text{diag}(\tau^{2,k}_{\gamma^k_{1:p}})\right) \\
\sigma^2_k &\sim \mathcal{IG}a(a,b) \\
\tau^{2,k}_{\gamma^k_{1:p}}|\gamma^k_{1:p} &\overset{\text{i.i.d.}}{\sim} \mathcal{E}x(\lambda^2/2)
\end{aligned}$$

where $\mathcal{N}_l(\mu, \Sigma)$ is the $l-$dimensional normal distribution of mean μ and covariance Σ. Note that, to simplify notation, when $l = 1$ we drop the subscript. $\mathcal{IG}a(a,b)$ is the Inverse-Gamma distribution of mean $b/(a-1)$ $(a > 1)$. $\mathcal{E}x(a)$ is the exponential distribution of mean $1/a$. The smoothness parameter λ controls the tail decay, as we can see from Figure 4.2 and is ultimately responsible to shrink the more weakly related regularization parameters to 0. Whilst in our discussion we assume λ is given, Park and Casella (2008) have shown, in a non-mixture Bayesian framework with known $\gamma_{1:p}$, that the Lasso parameter can be chosen by marginal maximum likelihood or using an appropriate hyperprior.

Comments on Bayesian Lasso

Even if the mode of the posterior, assuming a the double-exponential prior, is equivalent to the Lasso estimate, the marginal regularisation behaviour induced by the Bayesian version of the Lasso is qualitatively quite different from the one it shows in the frequentist setting. This is because, in the Bayesian approach, inference can be carried out with unified and computationally efficient sampling schemes, see section 4.3.2. The sampling based approach provides richer information on the posterior of a regression coefficient and adequately reflects the uncertainty in estimating a parameter to be close to zero. From the resulting samples, the posterior median or the posterior mean can easily be derived as point estimates, but computing the posterior mode is more difficult. Since the posterior for the regression coefficients will typically be skewed, the posterior mean and median will not coincide with the posterior mode and, as a consequence, will always be different from zero. Still, the concentration around zero will be quite high for redundant

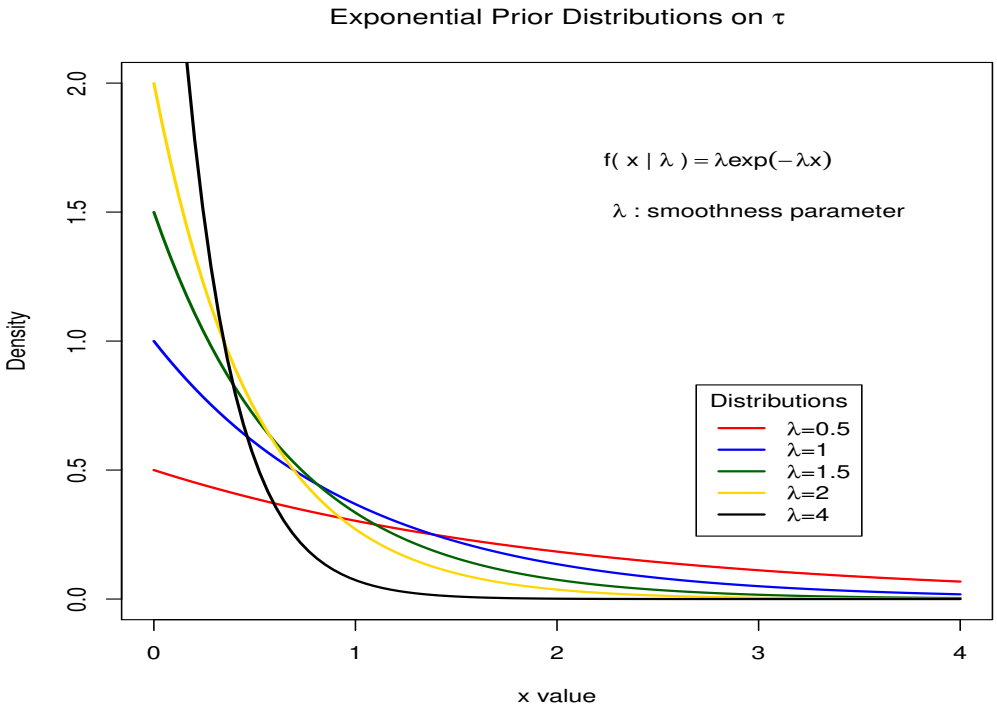

Figure 4.2: Exponential distribution. We can see the sensitivity of the tail decay to the smoothness parameter λ.

parameters so that coefficients may well be deemed to be equal to zero from a practical perspective.

Note also that, even if we will not be aiming to do this in the present discussion, the full conditional distributions of $\beta_{1:p}^k$, σ_k^2 and $\tau_{1:p}^{2,k}$ are still easy to sample, therefore estimating regularised regression coefficients and tuning the shrinkage parameter can be done simultaneously. In this context, the sensitivity of the results to the parameters can be measured via corresponding marginal posteriors.

As a further remark, although it seems to be quite restrictive to assume that all $\tau_{1:p}^k$ are i.i.d., the resulting class of regularisation priors is still large. Quite different classes of priors are obtained modifying the prior specifications, for examples see Fahrmeir et al. (2009).

Finally, while the Bayesian Lasso can effectively regularise the regression coefficients $\beta_{1:p}$ and provide an indication of which covariates are relevant, we also want

to address the regression model uncertainty and would prefer to make explicit the variable selection decision i.e. whether or not to have the covariate x_d with linear effect $\beta_d^k x_d$ as part of the linear predictor $\boldsymbol{x}'\boldsymbol{\beta}^k$.

4.2.2 Variable Selection

The frequentist version of the Lasso provides a straightforward method for variable selection by identifying the non-important predictor variables as those variables whose $\hat{\beta}_d^k = 0$. In the Bayesian version of the Lasso approach, under the absolutely continuous (w.r.t. Lebesgue measure) double-exponential prior distribution, the prior probability of the event $\{\beta_d^k = 0\}$ is zero. Thus, the posterior probability of such an event must also be zero. To overcome this problem, we are required to explicitly allocate prior probability mass to these events, $\{\beta_d^k = 0\}$ in order for posterior inferences about events to be coherent.

As we have seen in section 2.4.2, confronted with similar problems George and Mcculloch (1997); Tadesse et al. (2005); Kim et al. (2006); Schäfer and Chopin (2011) have proposed an effective solution by once again specifying a convenient prior for the selection indicator γ_d^k. Placing prior mass on the event $\{\beta_d^k = 0\}$ is equivalent to assigning a prior distribution to the space of two alternative regression models: one which includes the d^{th} covariate and the other one which excludes it. This can be done in a Bayesian framework using the latent indicator variable, $\gamma_{1:p}^k$, where $p(\gamma_d^k = 1)$ corresponds to prior probability of the model including the variable x_d and $p(\gamma_d^k = 0)$ indicates to probability the alternative event.

We adopt this solution and specify selection priors that fit into the mixture framework of regularised regressions. A suitable prior for γ_d^k is the Bernoulli distribution $\mathcal{B}e(\phi)$ which, under the assumption that $\gamma_{1:p}^k$ are independent, yields

$$\gamma_{1:p}^k \sim \prod_{d=1}^{p} \phi^{\gamma_d^k} (1-\phi)^{1-\gamma_d^k}. \tag{4.4}$$

Note that setting $\phi = 1/2$ means that when considering whether to include the variable x_d we do not have any prior information and the two alternative models,

$\gamma_d^k = 1$ and $\gamma_d^k = 0$, are equally likely. Only after having observed the data, the important predictor variables can then be identified by examining the marginal posterior inclusion probabilities.

We should also point out the level of the flexibility of the mixture model. By making $\gamma_{1:p}^k$ cluster specific, each regression curve can be a function of its own different set of covariates. On the other hand, the combinations of competing models to be evaluated grows exponentially with the number of explanatory variables and clusters, $K2^p$. In theory, for the given prior, we could compute the posterior probability of each model before selecting the best one. In practice, it is evident that a full exploratory search is unfeasible and we need to incorporate a selection procedure into the sampling algorithm.

In section 4.3 we discuss how to construct a stochastic sampler which will allow us to generate samples from the marginal posterior distribution of $\pi(\gamma_{1:p}^k)$. This solution will provide a viable computational approach for addressing model uncertainty while preserving the regularization properties induced by the Bayesian Lasso methodology. We first need to derive the posterior distribution of the parameters of interest.

4.2.3 Posterior Distribution

As we can see in Figure 4.3, we have been able to formulate a Bayesian hierarchical representation of the mixture of regressions model. We have also discussed in the previous section how the desired properties of robustness and parsimony are achieved by specifying convenient priors for the relevant parameters. We now derive the posterior of interest that will allow us to draw inference on the cluster membership of each observation and on the contribution of each variable.

Using a synthetic notation to indicate the unknown parameters of the model $\boldsymbol{\psi} = (w_{1:K}, \sigma_{1:K}, \boldsymbol{\beta}_{1:K}, \boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}_{1:p}^2)$, and the fixed, assumed known, hyperparameters of the model $\boldsymbol{h} = (a, b, \lambda, \phi, d, \delta)$, we can say that, after observing the covariates $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ and the responses $\boldsymbol{y} = (y_1, \ldots, y_n)$, the posterior distribution of

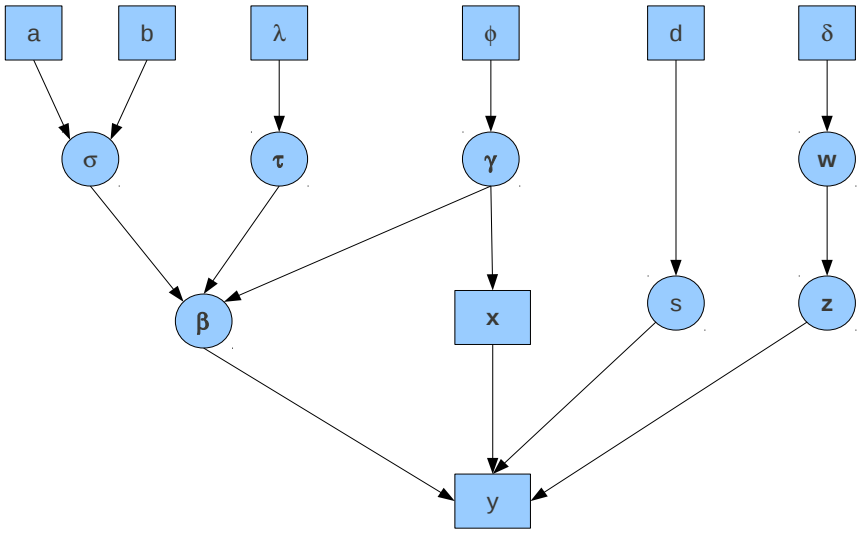

Figure 4.3: Directed Acyclic Graph (DAG) showing the hierarchical structure of the priors on the parameters of the proposed mixture model. We have drawn a square box around hyperparameters considered to be a known constant, a circle to indicate an latent variables that need to be estimated, and a rectangular box to indicate observed data. The arrows indicate the conditional dependence structure of the model.

$\boldsymbol{\psi}$ is

$$\pi(\boldsymbol{\psi}|\boldsymbol{x}, \boldsymbol{y}) \propto L(\boldsymbol{y}; \boldsymbol{x}, \boldsymbol{\psi})\, p(\boldsymbol{\psi}|\boldsymbol{h}) \tag{4.5}$$

where $L(\boldsymbol{y}; \boldsymbol{x}, \boldsymbol{\psi})$ is the likelihood function and $p(\boldsymbol{\psi}|\boldsymbol{x})$ the prior distributions we have previously defined.

Since our main focus is to draw inference on the cluster membership of the observations and identify the relevant explanatory variables, we remove as many other variables as possible. We integrate out the parameters $\boldsymbol{\beta}_{1:K}$, $\sigma_{1:K}$ and $w_{1:K-1}$ in (4.5)

$$\pi(z_{1:n}, \boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}|\mathcal{D}_n) = \int \pi(z_{1:n}, \boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}, \boldsymbol{\beta}_{1:K}, \sigma_{1:K}, w_{1:K-1}|\mathcal{D}_n)\, d(\boldsymbol{\beta}_{1:K}, \sigma_{1:K}, w_{1:K-1})$$

and obtain the marginal posterior density of interest up to a normalizing constant

$$\pi(z_{1:n}, \boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}|\mathcal{D}_n) \propto \prod_{k=1}^{K}\left[\xi_j(s^k_{1:n}, \gamma^k_{1:p}, \tau^{2,k}_{1:p}|\widetilde{\mathcal{D}}_k)\left\{\prod_{i=1}^{n}\varphi(s^k_i; d/2, d/2)\right\}\times \left\{\prod_{d:\gamma^k_d\neq 0}\varphi(\tau^{2,k}_d; 1, \lambda^2/2)\right\}\right]\frac{\prod_{k=1}^{K}\Gamma(\delta+n_k)}{\Gamma(\sum_{k=1}^{K}[n_k+\delta])} \tag{4.6}$$

where $\Gamma(\cdot)$ is the gamma function, $\varphi(x; a, b)$ is the Gamma density of mean a/b, $n_k = \sum_{i=1}^{n}\mathbb{I}_{\{k\}}(z_i)$ the number of observations assigned to the k^{th} cluster, $\widetilde{\mathcal{D}}_k$ is the collection of observations assigned to the k^{th} cluster. Given $z_i = k$, we can derive

$$\xi_k(s^k_{1:n}, \gamma^k_{1:p}, \tau^{2,k}_{1:p}|\widetilde{\mathcal{D}}_k) = \frac{|V^*_k|^{1/2}\Gamma(a^*_k)b^a(b^*_k)^{(-a^*_k)}}{|V_k|^{1/2}\pi^{n_k/2}\Gamma(a)}$$

with

$$\begin{aligned}
V_k &= \text{diag}(1, \tau^{2,k}_{\gamma^k_{1:p}}) \\
V^*_k &= \left(\text{diag}(\tau^{2,k}_{\gamma^k_{1:p}})^{-1} + x'_{\gamma^k_{1:p}}\Sigma^{-1}_{s^k_{1:n}}x_{\gamma^k_{1:p}}\right)^{-1} \\
m^*_k &= V^*_k(x'_{\gamma^k_{1:p}}\Sigma^{-1}_{s^k_{1:n}}y^k) \\
a^*_k &= a + n_k/2 \\
b^*_k &= b + \left((y^j)'\Sigma^{-1}_{s^k_{1:n}}y^k - (m^*_k)'(V^*_k)^{-1}m^*_k\right)/2
\end{aligned}$$

where $\Sigma_{s^k_{1:n}} = \text{diag}(s^k_1, \ldots, s^k_n)$.

With conjugate priors, the marginal posterior distribution of model parameters $(\boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p})$ and allocation variables $z_{1:n}$ are available in closed form. Nonetheless, sampling from the posterior distribution is the only viable approach that enables us to make inference on arbitrary functionals of the unknown variables. Given the high dimensionality of the posterior distribution we will require an efficient simulation methodology.

Note that if we were interested in simulating the regression coefficients $\boldsymbol{\beta}_{1:K}$ and the regression variance $\sigma^2_{1:K}$ we could sample the joint full conditional. Given

that the conjugate joint prior is a normal inverse-gamma, from Bayes' theorem the joint posterior given $z_i = k$ is

$$\pi(\boldsymbol{\beta}^k_{\gamma^k_{1:p}}, \sigma^2_k | \mathcal{D}_n, \tau^{2,k}_{1:p}, \gamma^k_{1:p}, s^k_{1:n}) \propto L(\boldsymbol{y}; \boldsymbol{x}, \boldsymbol{\psi})\, p(\boldsymbol{\beta}^k_{\gamma^k_{1:p}}, \sigma^2_k)$$

which yields

$$\pi(\boldsymbol{\beta}^k_{\gamma^k_{1:p}}, \sigma^2_k | \mathcal{D}_n, \tau^{2,k}_{1:p}, \gamma^k_{1:p}, s^k_{1:n}) \propto \quad (\sigma^2_k)^{(-a^*_k + \frac{\gamma^k_{1:p}}{2} + 1)} \;\times \exp\left\{ - \frac{(\boldsymbol{\beta}^k_{\gamma^k_{1:p}} - m^*_k)'(V^*_k)^{-1}(\boldsymbol{\beta}^k_{\gamma^k_{1:p}} - m^*_k) + 2\, b^*_k}{2\, \sigma^2_k} \right\}.$$

Given the allocations $w_{1:K-1}$ we would be able to sample the posterior of regression coefficients and regression variance for all components but this is outside the scope of the thesis. Nonetheless it constitutes an interesting route for further work even if the convergence of the MCMC is likely to be slowed down.

4.3 Simulation Methodology

Once we have been able to explicitly calculate the posterior distribution of the parameters of interest up to a constant, we then want to sample from this distribution in order to approximate quantities like expected values, which in our case are the classification probability of each observation and the marginal probability of inclusion of each variable.

The main tool available to Bayesian inference for sampling from a target distribution, such as π in (4.6), are the Monte Carlo methods we introduced in chapter 2. Among these, Markov chain Monte Carlo (MCMC) is the approach more frequently followed to draw inference on mixture models, see Robert and Casella (2004) for a complete review. In line with this approach, here we implement some recently proposed algorithms that have been developed to suit scenarios like ours, involving both an high dimensional model and complex patterns of dependence between parameters.

We should first note that, within the mixture modelling literature, there has

been work done on perfect sampling and direct sampling, making use of the full conditional distributions. For example, Mukhopadhyay and Bhattacharya (2011) proposed a perfect sampling methodology for fitting mixture models with either known or unknown number of components and applied this technique to both conjugate and to non-conjugate cases. Fearnhead and Meligkotsidou (2007) instead proposed a direct sampling method that returns independent samples from the true posterior. Unfortunately, the described algorithms have either limited applicability even for simple real life problems.

Considering that our proposed model allows for a random number of covariates, we need an algorithm flexible enough to explore a parameter space whose dimension is itself a random variable. Confronted with similar problem, Schäfer and Chopin (2011) used a Sequential Monte Carlo (SMC) algorithm, originally described by Del Moral et al. (2006), to adaptively sample from a binary distribution. Using the variable selection problem in linear regression as test case, they showed that even in difficult circumstances, with hundred of covariates, the Sequential Monte Carlo method can outperform standard techniques based on simple Markov chain exploration.

In light of recent work presented by Andrieu et al. (2010), we adopt a Particle Markov chain Monte Carlo (PMCMC) simulation procedure which combines MCMC and SMC methods and takes advantage of the strengths of both. The key feature of PMCMC algorithms is that they are in fact exact approximations of idealised MCMC algorithms, while they use sequential Monte Carlo methods to build high dimensional proposal distributions. On the other hand, compared to stand alone SMC, PMCMC sampling is less likely to suffer from the path depletion problem. More precisely, here we implement a particle Metropolis-within-Gibbs (PMWG) algorithm which is effective in situations where using the prior distribution of the underlying latent process as the proposal distribution is the only known practical solution.

4.3.1 Sampling Procedure

We can effectively achieve the result of simulating from the posterior distribution (4.6), following a two stage procedure. In the preliminary stage, we initialise the algorithm by sampling plausible parameters values from the corresponding priors and generate various particles that represent possible cluster assignments of the observed data points. In the subsequent stage, which should be repeated until convergence, we alternate a conditional SMC step, which produce a likely labelling of the data, and a Metropolis step, which updates the error estimate and the other parameters of interest.

Essentially, the sampling procedure consists in

- **Stage I:** Initialise the algorithm. Sample $\boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}$ from the respective priors. Run the SMC algorithm, as described in section 4.3.2, storing all the N particles labels $\boldsymbol{z}_{1:n} = z^1_{1:n}, \ldots, z^N_{1:n}$ and their genealogy $\boldsymbol{a}_{1:n-1} = (a^1_{1:n-1}, \ldots, a^N_{1:n-1})$ (defined below). Sample one particle index $t \in \{1, \ldots, N\}$ according to the normalized weights $\overline{W}^1_n, \ldots, \overline{W}^N_n$ (defined below).

- **Stage II:** Repeat the following steps till convergence

  1. Run the conditional SMC algorithm, as described in section 4.3.3.
  2. Sample $t \in \{1, \ldots, N\}$ according to new weights $\overline{W}^1_n, \ldots, \overline{W}^N_n$. Store $z^t_{1:n}$ and $b^t_{1:n}$ (defined below).
  3. Given $z^t_{1:n}$, update the current values of $\boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}$ following the MCMC steps described in section 4.3.4.

It is worth pointing out that, as we iterate through the simulation algorithm, the cluster structure evolves with the choice of variables and we should appreciate the fact that the variable selection, in the context of clustering, is much more complicated than in the standard classification or regression analysis.

We should also be aware of label switching problem which is a common issue when estimating the parameters of a Bayesian mixture model. In our implementation we adopt the practical solution of assuming that we already have a sensible

cluster assignment that we can use as a reference to guide the relabelling process after every MCMC iteration. More precisely, in the real life application we are going to discuss, we will permute all possible labelling combinations of the components and choose the one that maximizes the adjusted rand index computed with respect to cluster assignment proposed by the penalised t mixture model.

4.3.2 Sequential Monte Carlo Algorithm

With SMC methods we indicate a general class of algorithms that use a set of weighted particles to recursively approximate a sequence of distributions of increasing dimension. It has been originally introduced to suit situations with incoming observations, where any inferential statement has to be continuously updated. Nonetheless, it has demonstrated to be highly effective also in static problems like mixture models and it has become an integral part of PMCMC.

Before illustrating how the SMC algorithm is exploited in our sampling procedure, we refer to the appendix 4.A and to the work of Del Moral et al. (2006); Andrieu et al. (2010); Doucet et al. (2000) for a more detailed review of the different sampling methods we implement in the following section. In particular, we assume the reader is familiar with Sequential Importance Sampling (SIS), appendix 4.A.3.

Sampling Cluster Labels

The SMC method allows us to simulate from the conditional posterior distribution of the latent label indicator variables $\pi_n(z_{1:n}|\boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}, \mathcal{D}_n)$. Following Algorithm 4.1, we first initialize $\boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}$ by sampling their respective priors, and then alternate sequential importance sampling and resampling steps.

More explicitly, the sequential importance sampling targets the full conditional density of the latent labels variables $z_{1:i}$ which, after the first $1, \dots, i$ data points, is

$$\pi_i(z_{1:i}|\boldsymbol{s}_{1:i}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}^2_{1:p}, \mathcal{D}_i) \propto \left[\prod_{k=1}^{K} \xi_k(s^k_{1:i}, \gamma^k_{1:p}, \tau^{2,k}_{1:p}|\widetilde{\mathcal{D}}^{(i)}_k)\Gamma(\delta + n^{(i)}_j)\right] \Big/ \Gamma(\sum_{k=1}^{K}(n^{(i)}_k + \delta))$$

where $\widetilde{\mathcal{D}}_k^{(i)}$ denotes the data allocated to the k^{th} cluster out of the first i observations and

$$n_k^{(i)} = \sum_{l=1}^{i} \mathbb{I}_{\{k\}}(z_l)$$

their total number.

Adaptive Resampling

Without supervision, sequential importance sampling can incur the problem of weight degeneracy. As new incoming observations are fed into the algorithm, the variance of $\omega_n(x_{1:n})$ typically increases at an exponential rate until all the mass concentrates on one single particle, leaving the remaining particles with weights tending to zero.

To avoid spending a large computational effort to update trajectories whose contribution to the final estimate is negligible, we execute a resampling step with the intention of replacing the unpromising lowest weighted particles with new particles that hopefully lie in regions of high target density. The exact procedure consists in sampling N particles from the approximated target distribution $\hat{\pi}_n(x_{1:n})$ to obtain N new particles which will then be equally weighted. At the subsequent iteration of the importance sampling step, the new weights will be simply equal to the incremental weights, $W_{1:n+1} = \alpha_{n+1}(x_{1:n+1})$.

On the other hand, if we iterate several times the weighting and resampling procedure, we will rapidly deplete the number of distinct particles and the accuracy of the full conditional density estimation will suffer because the paths become very similar. In particular for $i << n$ the marginal distribution $\hat{\pi}_n(x_{1:i})$ will only be approximated by a few if not a single unique particle. This is due to the fact that only the variables $\{X_n^j\}$ are sampled at time n, whereas the path values $\{X_{1:n-1}^j\}$ are not rejuvenated.

To find a balance between weights degeneracy and path degeneracy, Liu (2001) and Del Moral et al. (2011) among others, suggest to resample only when the variance of the unnormalized weights is above a fixed threshold. In the solution

we adopt, the threshold is a function of the Effective Sample Size (ESS)

$$ESS = \left(\sum_{j=1}^{N} (W_n^j)^2 \right)^{-1}$$

which takes values between 1 and N and, as described in Algorithm 4.1, we resample only when it is below $ESS < N/2$. To fully appreciate the effect of introducing this rule, in the experimental section 4.4.2 we tested and compared the two versions of the algorithm, with and without adaptive resampling, and show the different impact they have on weights dispersion and paths diversity.

It should be noted here that executing the resampling step only when the condition $ESS < N/2$ is satisfied, does not alter the property of the algorithm that still returns an unbiased empirical estimate of the target distribution, since the estimate of the normalising constant is unbiased, as noted in a personal communication by Andrieu C. and Whiteley N..

Algorithm 4.1 Sequential Monte Carlo Algorithm

Step 1. Sample N labels, $z_1^1, \ldots, z_1^N$, from $\pi_1(z_1|\cdots)$ and set the corresponding weights $W_1^j = 1$ for $j = 1, \ldots, N$.

Step 2. For $i = 2, \ldots, n$ repeat the following

1. If $ESS < N/2$, for each $j = \{1, \ldots, N\}$ resample $a_{i-1}^j \in \{1, \ldots, N\}$ using the discrete distribution

$$\overline{W}_{i-1}^j = \frac{W_{i-1}^j}{\sum_{g=1}^N W_{i-1}^g}.$$

   otherwise keep all the current particles by $a_{i-1}^j = j$ for $j \in \{1, \ldots, N\}$.

2. sample, for each $j \in \{1, \ldots, N\}$, a label z_i^j from $\pi_i(z_i|\cdots)$ where

$$\pi_i(z_i|\cdots) = \frac{\xi_{z_i}(s_{1:i}^{z_i}, \gamma_{1:p}^{z_i}, \tau_{1:p}^{2,z_i}|\widetilde{\mathcal{D}}_{z_i})\Gamma(\delta + 1 + n_{z_i}^{(i-1),a_{i-1}^j})}{\sum_{z_i=1}^K \xi_{z_i}(s_{1:i}^{z_i}, \gamma_{1:p}^{z_i}, \tau_{1:p}^{2,z_i}|\widetilde{\mathcal{D}}_{z_i})\Gamma(\delta + 1 + n_{z_i}^{(i-1),a_{i-1}^j})}$$

   and $n_k^{(i-1),a_{i-1}^j} = \sum_{i=1}^{i-1} \mathbb{I}_{\{k\}}(z_i^{(a_{i-1}^j)})$. Set $z_{1:i}^j = (z_{1:i-1}^{a_{i-1}^j}, z_i^j)$.

3. Set, for each $j \in \{1, \ldots, N\}$

$$W_i^j = \frac{\sum_{z_i=1}^K \left[\prod_{k=1}^K \xi_k(s_{1:i}^k, \gamma_{1:p}^k, \tau_{1:p}^{2,k}|\widetilde{\mathcal{D}}_k^{(i)})\Gamma(\delta + n_k^{(i),j})\right] \Big/ \Gamma(\sum_{k=1}^K (n_k^{(i),j} + \delta))}{\left[\prod_{k=1}^K \xi_k(z_{1:i-1}^k, \gamma_{1:p}^k, \tau_{1:p}^{2,k}|\widetilde{\mathcal{D}}_k^{(i-1)})\Gamma(\delta + n_k^{(i-1),j})\right] \Big/ \Gamma(\sum_{k=1}^K (n_k^{(i-1),j} + \delta))}$$

   and $i = i + 1$.

4.3.3 Conditional Sequential Monte Carlo Algorithm

The conditional SMC algorithm we iterate in the second stage of our sampling procedure is essentially the SMC algorithm described in Section 4.3.2 except it preserves the path of one particle.

To describe the algorithm, we need to introduce a sequence of indexes $b_{1:n}^t \in \{1,\ldots,N\}^n$ to represent the genealogy of the t^{th} particle for $t \in \{1,\ldots,N\}$. Once we have set $b_n^t = t$, the genealogy of t^{th} particle can then be defined recursively $b_i^t = a_{i-1}^{b_i^t}$ for $i = 1,\ldots,n-1$ where the $\boldsymbol{a}_{1:n-1} = (a_{1:n-1}^1,\ldots,a_{1:n-1}^N)$ are the recorded samples from the previous iteration of the SMC algorithm.

As we can see from the Algorithm 4.2, the sampling sequence is similar to what is implemented in a standard SMC algorithm except that one randomly chosen particle t with its ancestral lineage $b_{1:n}^t$ is fixed and ensured to survive, whereas the remaining $N-1$ particles are regenerated as usual.

Algorithm 4.2 Conditional Sequential Monte Carlo Algorithm

Step 1. Sample $1-N$ labels z_1^j from $\pi_1(z_1|\cdots)$, for $j = 1,\ldots,N$ while $j \neq b_1^t$ (i.e. excluding $j = b_1^t$), and set all the weights $W_1^j = 1$ for $j = 1,\ldots,N$.

Step 2. For $i = 2,\ldots,n$ repeat the following

1. If $ESS < N/2$, for each $j \in \{1,\ldots,N\}$ except $j = b_i^t$, resample $a_{i-1}^j \in \{1,\ldots,N\}$ using the discrete distribution

$$\overline{W}_{i-1}^j = \frac{W_{i-1}^j}{\sum_{g=1}^N W_{i-1}^g}.$$

   otherwise keep all the current particles by $a_{i-1}^j = j$.

2. Sample z_1^j from $\pi_i(z_i|\cdots)$ for each $j \in \{1,\ldots,N\}$ except $j = b_i^t$, and update the corresponding path $z_{1:i}^j = (z_{1:i-1}^{a_{l-1}^j}, z_i^j)$.

3. Set W_i^j as for the SMC algorithm, for each $j \in \{1,\ldots,N\}$, (this includes the fixed particle $j = b_i^t$)

4.3.4 Markov Chain Monte Carlo Steps

In the SMC algorithm we sample from the posterior distribution of the latent label indicator variable Z and propose a cluster assignment of each observation. With the MCMC steps our objective is to update the other parameters of the mixture model that control the regression error distribution, the regularization of the regression coefficients and the variable selection process. For a more complete review of MCMC methods we refer to the section of the appendix 4.A.1, where the essential aspects of the methods are discussed. Here we just remark this; the fact that executing the MCMC moves allows us to explore neighbouring models which are copies of the current model where just a few components are altered.

The MCMC kernels we adopt suit the requirements of our simulation procedure and provide an effective way to sample from complex probability measures. That is, they are more likely to find regions of high probability measure than simple importance functions. More precisely, the transition kernels we implement are often referred to as metropolised Gibbs samplers, since they proceed component-wise as the classical Gibbs sampler, which sequentially draws each component from the full marginal distribution, see appendix 4.A.2. Sill, the sampling procedure is qualified as *metropolised* because it accepts or refuses the proposed move according to a Metropolis-Hastings step. Essentially, once we have generated a candidate value x_i^* from a proposal distribution $q(\cdot|x_{i-1})$, the basic idea of the Metropolis-Hastings algorithm is to accept the move with probability $\min\{1, A\}$ where A is

$$A = \frac{\pi(x_i^*)q(x_{i-1}|x_i^*)}{\pi(x_{i-1})q(x_i^*|x_{i-1})}$$

otherwise x_i^* is rejected and we stay at $x_i = x_{i-1}$.

One critical decision we need to take is to fix the average size of the proposed moves which depends on hyperparameters we control, see the work of Roberts et al. (1997). Large step proposals improve the mixing properties of the chain and help to escape from the attraction of local modes. The inconvenience in this case might be that the acceptance rate becomes excessively low since we blindly propose arbitrary points in the sampling space. If, instead, we change one component at

a time with smaller steps, we will quite often accept the move, but it is unlikely that we will efficiently explore the entire parameter space. To clarify the concept, we document the direct relation between the step length and the acceptance rate in the experimental section 4.4.2.

Having discussed the general MCMC approach and the specific version of the Gibbs sampler we intend to use, we should now see how this is implemented in practice in our model. Let us first note that the Particle Gibbs Metropolis-Hastings update we propose is defined on an extended space which includes the N label particles $\boldsymbol{z}_{1:n} = z_{1:n}^1, \ldots, z_{1:n}^N$, their genealogy $\boldsymbol{a}_{1:n-1} = (a_{1:n-1}^1, \ldots, a_{1:n-1}^N)$, the number of clusters K, and the vector of parameters and hyperparameters of interest $\boldsymbol{\theta} = (\boldsymbol{s}_{1:n}, \boldsymbol{\gamma}_{1:p}, \boldsymbol{\tau}_{1:p}^2)$,

$$\overline{\pi}(K, \boldsymbol{z}_{1:n}, \boldsymbol{a}_{1:n-1}, \boldsymbol{\theta}|\mathcal{D}_n) = \frac{\pi(z_{1:n}^k, \boldsymbol{\theta}|\mathcal{D}_n)}{N^n} \frac{\psi^{\boldsymbol{\theta}}(\boldsymbol{z}_{1:n}, \boldsymbol{a}_{1:n-1})}{\pi_1(z_1^{b_1^k}|\cdots) \prod_{i=2}^n \overline{W}_{i-1}^{b_{i-1}^k} \pi_i(z_i^{b_i^k}|\cdots)}$$

and targets the probability density

$$\overline{E} = \{1, \ldots, N\}^{(n-1)N+1} \times \{1, \ldots, K\}^{nN} \times \mathbb{R}_+^{nK} \times \{0,1\}^{Kp} \times \left(\bigcup_{z=1}^{Kp} \mathbb{R}_+^z \right).$$

The single MH steps, that separately update the elements of $\boldsymbol{\theta}$, are then as follow.

Step 1: Update $\boldsymbol{\tau}_{1:p}^2$

To update the $\boldsymbol{\tau}_{1:p}^2$, given all the other variables are fixed, we can use the following procedure. For each $k \in \{1, \ldots, K\}$, assuming $|\gamma_{1:p}^k|_1 > 0$, sample for each d where $\gamma_d^k = 1$

$$(\tau_d^{2,k})^* = \tau_d^{2,k} \exp\{\nu_\tau \, N_d\} \tag{4.7}$$

with $\nu_\tau > 0$ a user-set parameter and $N_d \sim \mathcal{N}(0,1)$, independent for each d. Accept all the $(\tau_d^{2,k})^*$ with probability

$$\min\left\{1, \frac{\xi_k(s_{1:n}^k, \gamma_{1:p}^k, (\tau_{1:p}^{2,k})^* | \widetilde{\mathcal{D}}_k)}{\xi_k(s_{1:n}^k, \gamma_{1:p}^k, \tau_{1:p}^{2,k} | \widetilde{\mathcal{D}}_k)} \prod_{d;\gamma_d^k \neq 0} \frac{\varphi((\tau_d^{2,k})^*; 1, \lambda^2/2)(\tau_d^{2,k})^*}{\varphi(\tau_d^{2,k}; 1, \lambda^2/2)\tau_d^{2,k}}\right\}$$

otherwise keep the current $\tau_{1:p}^{2,k}$. The chain is reversible as a by-product from being a Metropolis-within-Gibbs algorithm. The chain leaves the extended target invariant using a simple adaptation of Theorem 5 of Andrieu et al. (2010). That the extended target admits the posterior of interest as an appropriate marginal follows from the fact that the SMC with adaptive resampling provides an unbiased estimate of the normalizing constant; see the decomposition of Arnaud and Le Gland (2009).

Step 2: Update $\boldsymbol{s}_{1:n}$

To update $\boldsymbol{s}_{1:n}$, given all the other variables are fixed, we can use the following procedure. For each $i \in \{1, \ldots, n\}$, $k \in \{1, \ldots, K\}$ propose

$$(s_i^k)^* = s_i^k \exp\{\nu_s N_i\} \tag{4.8}$$

where $\nu_s > 0$ is a user-set parameter (potentially different from the ν_τ above) and $N_i \sim \mathcal{N}(0,1)$, independent for each i. Note that $(s_{1:n}^k)^*$ features only one changed value from $s_{1:n}^k$. The proposed move then is accepted with probability

$$\min\left\{1, \frac{\xi_k((s_{1:n}^k)^*, \gamma_{1:p}^k, \tau_{1:p}^{2,k} | \widetilde{\mathcal{D}}_k)}{\xi_k(s_{1:n}^k, \gamma_{1:p}^k, \tau_{1:p}^{2,k} | \widetilde{\mathcal{D}}_k)} \prod_{i;\gamma_i^k \neq 0} \frac{\varphi((s_i^k)^*; d/2, d/2)(s_i^k)^*}{\varphi(s_i^k; d/2, d/2)s_i^k}\right\}$$

otherwise keep the current s_i^k.

Step 3: Update $\boldsymbol{\gamma}_{1:p}$

To update $\boldsymbol{\gamma}_{1:p}$, given all the other variables are fixed, we can use the following procedure. For each $d \in \{1, \ldots, p\}$, $k \in \{1, \ldots, K\}$ (i.e. propose to change only one element each time), if $\gamma_d^k = 0$ we propose $(\gamma_d^k)^* = 1$ and draw $(\tau_d^k)^*$ from its prior $(\mathcal{G}a(1, \lambda^2/2))$. The proposed move is accepted with probability

$$\min\left\{1, \frac{\xi_k(s_{1:n}^k, (\gamma_{1:p}^k)^*, (\tau_{1:p}^{2,k})^* | \widetilde{\mathcal{D}}_k)}{\xi_k(s_{1:n}^k, \gamma_{1:p}^k, \tau_{1:p}^{2,k} | \widetilde{\mathcal{D}}_k)}\right\}$$

otherwise we keep $\gamma_d^k = 0$. If $\gamma_d^k = 1$, we propose to set it to be zero, removing the corresponding $\tau_d^{2,k}$ and using the same expression as above to accept/reject (with the appropriate changes i.e. the proposed state here has fewer variables than the current model). In this proposal, we are adding or removing columns from our design matrix.

Note that this algorithm is best suited for scenarios similar to the ones we investigate in this thesis, where the number of components $K > 2$ and the number of data points $n \geq 40$ make the space to be sampled much bigger than the one for the explanatory variables. Before applying to real life data we test and observe its performance on simulated data.

4.4 Numerical Examples

The scope of this section is to verify, using simulated datasets, the properties of the mixture model we propose. Testing it in a controlled environment, we aim to highlight the main properties and possible faults of the model.

First we assess how the simulation procedure responds to different scenarios, its sensitivity to changes in prior hyperparameters and how to set the parameters that control the algorithm. In particular, we monitor the acceptance rate of the metropolised Gibbs sampling steps and the degeneracy of the weights and path diversity in the two cases: with resampling that is executed at every iteration and with adaptive resampling which is subject to ESS criterion being satisfied.

The performance of the proposed model is then discussed in terms of clustering accuracy, which means checking that homogeneous observations are correctly grouped together. At the same time we also interested to verify that only the truly informative variables are actually included in the model.

The model simulation procedure and its related sampling algorithms have been coded in Matlab; the code is available on request.

4.4.1 Simulation Settings

We assume one basic scenario that we then perturb to highlight the different properties of the model and different important aspects of the simulation procedure. In the standard scenario the parameters of the model have been randomly generated from the following priors

$$\begin{aligned}
w_{1:K-1} &\sim \mathcal{D}ir(2) \\
s_i^k &\sim \mathcal{G}a(2,2) \\
\gamma_{1:p}^k &\overset{\text{i.i.d.}}{\sim} \mathcal{B}e(1/2) \\
\tau^{2,k}_{\gamma_{1:p}^k} | \gamma_{1:p}^k &\overset{\text{i.i.d.}}{\sim} \mathcal{E}x(1/2) \\
\sigma_k^2 &\sim \mathcal{IG}a(2,4) \\
\beta^k_{\gamma_{1:p}^k} | \sigma_k^2, \tau^{2,k}_{\gamma_{1:p}^k}, \gamma_{1:p}^k &\sim \mathcal{N}_{|\gamma_{1:p}^k|_1}\left(0, \sigma_k^2 \text{diag}(\tau^{2,k}_{\gamma_{1:p}^k})\right)
\end{aligned}$$

and each data point is then sampled from the mixture

$$y_i \sim \sum_{k=1}^{K} w_k \mathcal{N}(x'_{\gamma_{1:p}^k, i} \beta^k_{\gamma_{1:p}^k}, s_i^k).$$

Each dataset we generate contains $n = 50$ paired observations sampled from a mixture of three components $K = 3$. More precisely, for each data point $i \in \{1, \ldots, n\}$ we first sample a label $z_i = k$ from $\mathcal{M}(1|w_1, \ldots, w_K)$ then sample the covariates $\boldsymbol{x}_i$ of dimension $p = 20$ from a centered Gaussian distribution whose dispersion depends on the cluster membership. We finally obtain the response

variable y_i by adding a random error s_i^k to the systematic component $\boldsymbol{x}'_{\gamma_{1:p}^k}\boldsymbol{\beta}^k_{\gamma_{1:p}^k}$.

The only parameters of the simulation algorithm we need to set are: the number of particles, say $N = 100$; the step length of the proposed MH move for τ, say $\nu_\tau = 2$; the step length of the error update, say $\nu_z = 3$, and also the number of repeats of the sampling procedure, say a few thousand.

4.4.2 Sensitivity of the Simulation Methodology

We first want to assess how the simulation procedure responds to changes in the step length of the MCMC moves and how the resampling option influences the weights and path degeneracy.

Acceptance Rate

The MCMC updating algorithms described in (4.7) and (4.8) depend on the parameters ν_τ and ν_s respectively. These parameters have to be set by the user and are important because they control the range of the steps the algorithm can take to explore the parameter space and ultimately determine the speed and mixing properties of the chain.

In Figure 4.4 we see that, as expected, the acceptance rate of both updates gradually decays as we increase the step length. This allows us to identify an interval of values for ν_τ and ν_s that corresponds to a target acceptance usually fixed between $0.20 - 0.60$.

Weights and Paths Degeneracy

The other important aspect of the simulation behaviour that we can partially control is the weight degeneracy. By introducing the adaptive resampling step we limit the risk of the empirical probability mass collapsing on a single particle. We are equally aware that resampling tends to replicate the most likely paths and might lead to an impoverished diversity of explored paths. This effect is also marginally alleviated by limiting the frequency of the resampling.

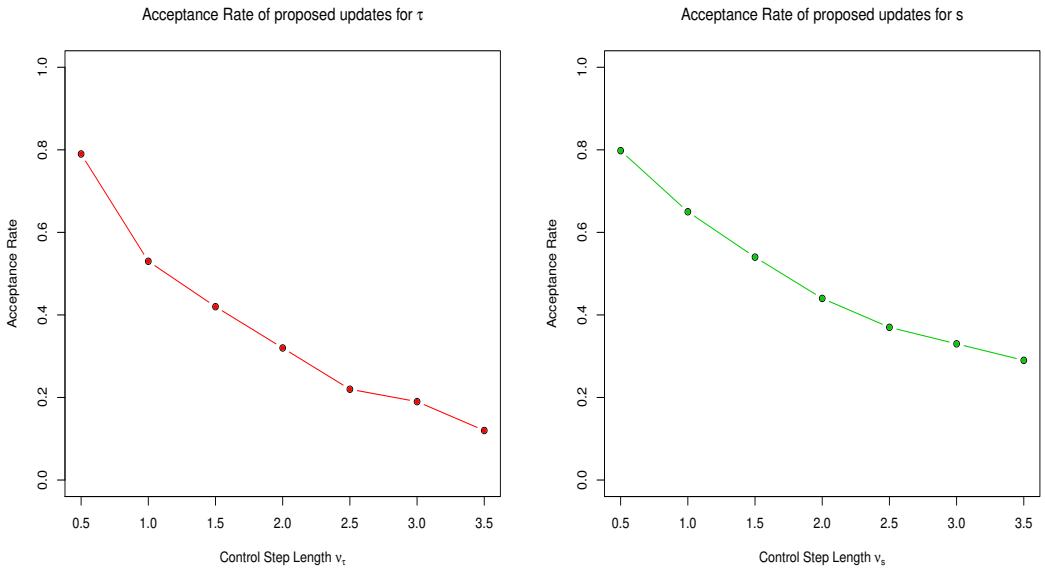

Figure 4.4: Acceptance rate as a function of step length. Left plot, acceptance rate of the MCMC move to update $\tau_{1:p}^{2,k}$ as a function of the control parameter ν_τ. Right plot, acceptance rate of the MCMC move to update $s_{1:n}^{k}$ as a function of the control parameter ν_s.

Figure 4.5 shows that, in our case, adaptive resampling ultimately is beneficial to preserve both path and weight diversity. We note in the left column that if we systematically resample after every new observation is processed, we end up fairly quickly with a single path that gets replicated for all N particle. As that happens, all particles become equally likely, $W^j = 1/N$ for each $j \in \{1, \ldots, N\}$, and $ESS/N = 1$ almost always. Figure 4.5 shows that, in our case, adaptive resampling ultimately is beneficial to preserve both path and weight diversity. We note in the left column that if we systematically resample after every new observation is processed, the paths diversity drops fairly quickly till the same path gets replicated for all N particle. As that happens, all particles become equally likely, $W^j = 1/N$ for each $j \in \{1, \ldots, N\}$, and $ESS/N = 1$ almost always.

With adaptive resampling, on the other hand, the degeneracy of weights and paths is maintained at a tolerable level. In the right column, we preserve a variety of paths that might have different likelihood as shown by the more disperse ESS

plot. Note also how in some instances no resample is performed for several runs and the number of particles remains stable as it is their weight. Even if at the end of every iteration of the sampling procedure we only need to store one single particle, it is important that we are able to preserve a richer variety of paths and consequently a more homogeneous weight distribution from which we can sample.

4.4.3 Model Performance

We suggested on more than one occasion that our main objective is to have a model that can simultaneously cluster the observations and select the relevant variables. To assess how accurately our model performs these tasks, we employ the same three indicators we introduced in section 3.5.

Clustering Accuracy

To test the clustering accuracy of the mixture of Lasso regressions with Student's t errors, we generate random datasets using the simulation settings described in 4.4.1. We then let the algorithm run and for each iteration we save one particle that represents one sample from the posterior distribution of the label indicator variables, $\pi(z_{1:n})$. Once we have collected enough samples we analyse the distribution of the Adjusted Rand Index Score over the sampled paths.

In Figure 4.6 we can see that the distribution is highly skewed towards 1, which means that most of the time the suggested clustering assignment perfectly matches the true clustering. In other words, we can say that the classification probability distribution we try to approximate $\hat{\pi}(z_{1:n})$ is fairly accurate and well representative of the observed data, at least in this example.

Variable Selection Accuracy

The other major point we want to investigate is the accuracy of the variable selection approach we implemented for the mixture of regressions. We would hope that the model identifies as many informative variables as possible, and at the

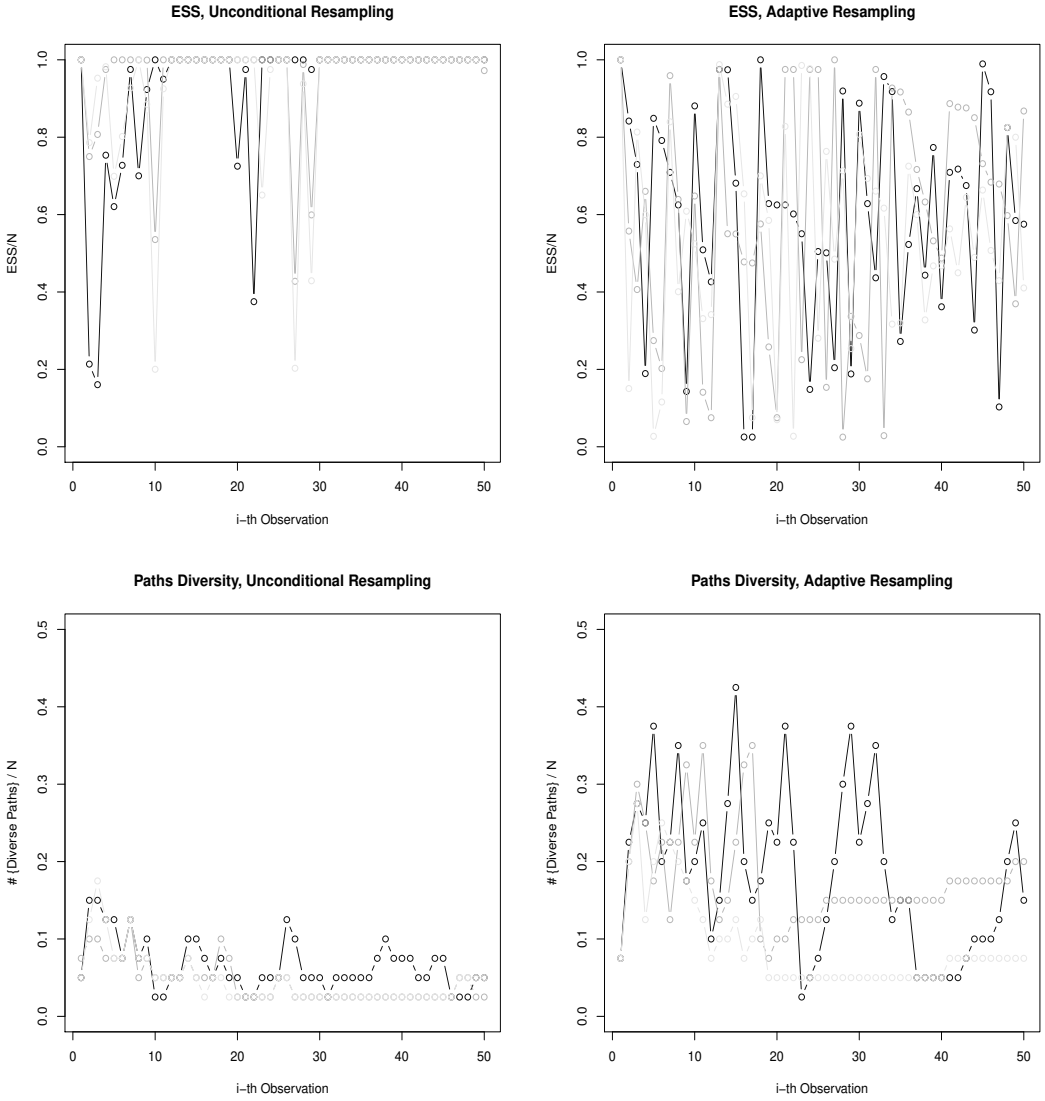

Figure 4.5: **Left Column:** Unconditional Resampling, we resample systematically every time a new observation is fed into the SMC algorithm. **Right Column:** Adaptive Resampling, we only resample whenever the ESS falls below a fixed threshold. **Top Row:** Weight Degeneracy, measured as ESS/N, where 1 means all particles have equal weight, and 0 means the entire probability mass is on one particle. **Bottom Row:** Path Degeneracy, measured as percentage of paths that remain different as we loop through the observations. Each line represents three separate repeats of the sampling procedure and darker lines correspond to earlier iterations.

same time is sufficiently parsimonious to exclude as many as possible of the noise variables.

In Figures 4.7 and 4.8 we look, as before, at the distribution over all MCMC

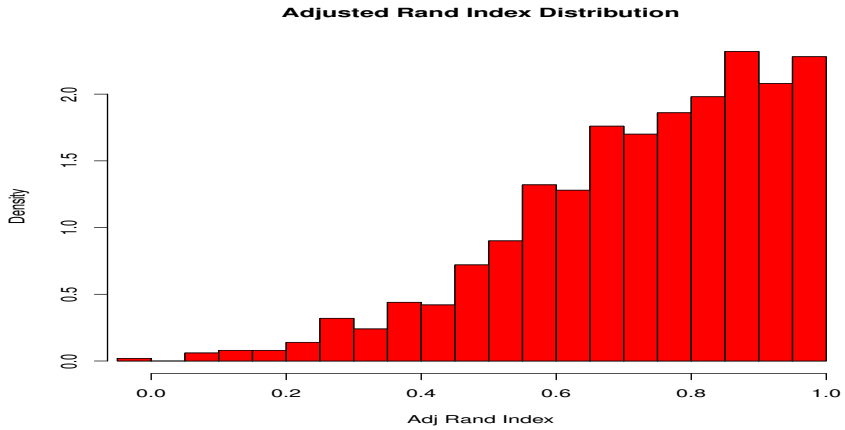

Figure 4.6: Adjusted Rand Index distribution. For every MC iteration we record the adjusted Rand Index score of the proposed cluster assignment versus the true clusters labels. Where a distribution centered around zero would be an indication of random assignment, the observed values give evidence that the model is successfully assigning most of the data points to the proper cluster.

iterations of the relevant indices, in this case the sensitivity and specificity indexes. Note that given the relatively small number of variables, $p = 20$, we should not be surprised to observe some very coarse distributions, since there are only so many informative or noise variables. In both plots it is evident that the overall variable selection accuracy is considerable. The sensitivity of the selection algorithm is fairly high, since most of the informative covariates are included and play a role in the regression curves. Conversely, the specificity index is equally good if not better, as very few noise variables are retained at all. We can explain the marginally lower sensitivity compared to the specificity, by noticing that the model is successfully parsimonious and achieves a satisfactory clustering performance even with only a smaller subset of the informative variables.

4.5 Conclusions

In this chapter we have studied the problem of clustering paired samples of input and output variables while attempting to identify the truly relevant covariates. Following a Bayesian approach we have proposed a mixture of Lasso regressions

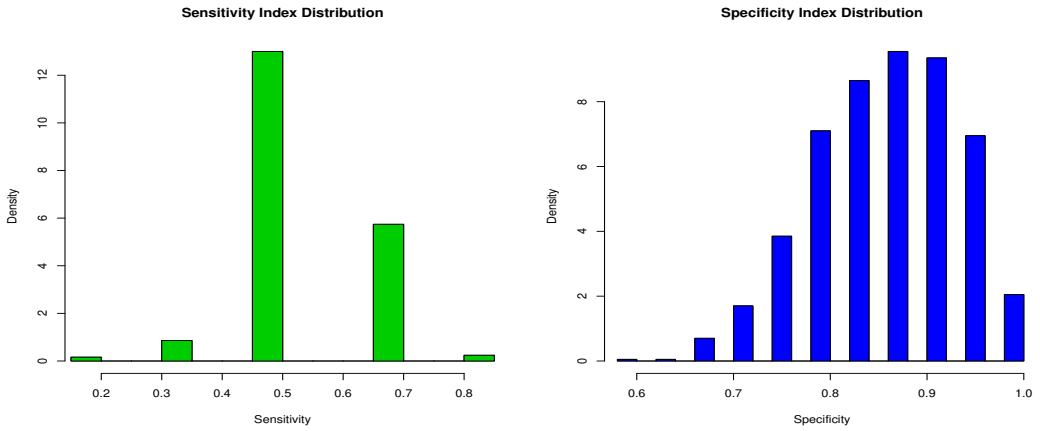

Figure 4.7: Variable Selection accuracy over all MC iterations. In the left plot we show the distribution of the Sensitivity index, i.e. the ability of the algorithm to identify the truly informative variables. In the right plot the Specificity index measures the accuracy of the model in isolating the non-informative variables. On the other hand, the right plot shows that the model is very precise in excluding the noise variables.

that respond to these requirements.

The model, by construction, admits a sparse solution which is achieved through a cluster specific binary vector that dictates which variables should be included and which variable should be excluded. To ensure we also obtain a robust solution, we allow for regression errors to be student's t distributed and implement a Bayesian Lasso approach that impose regularization on the regression coefficients.

Since the model is complex and high-dimensional, an efficient sampling routine is required to approximate the posterior distribution of the quantities of interest. We implemented a Particle Markov chain Monte Carlo simulation procedure that alternates a conditional Sequential Monte Carlo algorithm to sample from the posterior of the clustering labels, with a Metropolised Gibbs sampler that update the other relevant parameters conditional on the proposed cluster assignment.

We coded the estimation procedure in Matlab and used simulated data to asses how accurately the model clusters the observations and selects the right explanatory variables. We have first verified the sensitivity of the sampling algorithm to

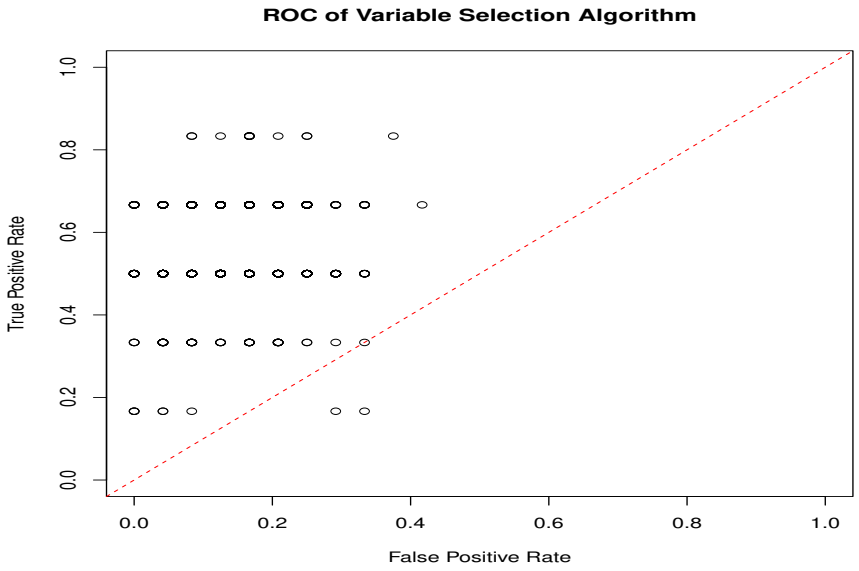

Figure 4.8: Receiver Operating Characteristic, this plots illustrates the possible risk that by including too many variables we could also have many noise variables slipping through. In reality we observe that our model is fairly accurate as it can identify and include the greater majority of informative variables with a very small error rate.

changes in the set parameters that control the step length of the update moves. Secondly we have demonstrated how we can contrast the weights and paths degeneracy in the SMC routine, using a trigger condition to execute the resampling step. We have then discussed in more detail the performance of the proposed model in likely standard scenarios.

Having verified that the model behaves and performs in line with our expectations, we can now rely on it to investigate the real life data problems we described in chapter 1.

As a final remark, we should say that the potential extension of the model would involve studying the case where the number of clusters K is unknown and removing the assumption of independence between the covariates.

4.A Appendix: Sampling Algorithms

For a more in depth discussion of Markov Chain Monte Carlo methods we refer the reader to the related work of Robert and Casella (2004, 2011); Doucet et al. (2000). Here we only review the essential notions and methods that are at the core of the simulation method we propose.

In chapter 2, we introduced the basics of Monte Carlo methods and Importance Sampling which we consider preliminary material for the following discussion.

4.A.1 Markov Chain Monte Carlo

We first describe the theory underlying MCMC methods, and then introduce the Gibbs sampler, an MCMC method which is particularly suited to solve mixture model inferential problems.

The MCMC methods provide an alternative way of approximating the integral $I(h)$ we defined in (2.20). We suggest an effective sampling procedure which relies on the construction of an ergodic Markov chain with stationary distribution π.

A Markov chain in discrete time and general state space $\mathcal{X}$, is a sequence of random variables $(X_0, X_1, \ldots)$, with $X_n \in \mathcal{X}$ for $n \geq 0$, which obeys the Markov Property. That is, given the current state X_n, the distribution of the next state X_{n+1} is independent of the past history of the chain, $(X_1, \ldots, X_{n-1})$. When this condition is verified, the distribution of the time homogeneous Markov chain $\{X_n\}$ on state space $\mathcal{X}$ is fully specified by the distribution of X_0 and by its transition kernel.

To construct a Markov chain which is $\pi-$invariant and ergodic, we need to define easily simulated transition probabilities $P(x, x^*)$ for $x, x^* \in \mathcal{X}$, such that

$$\int_{x \in \mathcal{X}} \pi(dx) P(x, x^*) = \pi(x^*).$$

Once we have run the Markov chain for a sufficiently long time, i.e. for large n, we assume that the chain has been able to exhaustively visit the state space of the support of π and that the distribution of X_n has become stationary. We can

then collect a sufficiently large number of sample N from the chain and use the empirical measure to approximate the target distribution π

$$\hat{\pi}(x) = \frac{1}{N}\sum_{j=1}^{N} \delta_{X^j}(x).$$

Therefore, due to the strong-law of large numbers for Markov Chains, see Meyn and Tweedie (1993), the MCMC estimate

$$I^{MCMC}(h(x)) = \int h(x)\,\hat{\pi}(x)dx = \frac{1}{N}\sum_{j=1}^{N} h(X^j)$$

converges almost surely to $I(h)$.

4.A.2 Gibbs Sampler

The Gibbs sampler is designed to handle multidimensional problems where the transition kernel is formed by the full conditional distributions.

Suppose that π is a $p-$dimensional density, known up to a normalizing constant, which is defined on $\mathcal{X}$, an open subset of $\mathbb{R}^p$, and denote $\pi(\boldsymbol{x}) = \pi(x_1, \ldots, x_p)$ the distribution we want to sample from.

As we can see in Algorithm 4.3, at each step the Gibbs sampler replaces the value of one of the variables with a value drawn from the distribution of that variable conditioned on the remaining variables being fixed. More precisely, x_d is replaced by a value drawn from the full conditional distribution $\pi_d(x_d|\boldsymbol{x}_{\setminus d})$ where x_d denotes the $d-$component of $\boldsymbol{x}$ and $\boldsymbol{x}_{\setminus d}$ denotes $x_1, \ldots, x_p$ but with x_d omitted.

Algorithm 4.3 Gibbs Sampling Algorithm

Step 1. Choose an initial state $x^0 \in \mathcal{X}$

Step 2. For $j = 1, \ldots, N$ repeat the following

Draw $X_1^j \sim \pi_1(x_1 | \boldsymbol{x}_{\setminus 1}^{j-1})$

$\vdots$

Draw $X_d^j \sim \pi_d(x_d | x_{1:d-1}^j, x_{d+1:p}^{j-1})$

$\vdots$

Draw $X_p^j \sim \pi_p(x_p | \boldsymbol{x}_{\setminus p}^j)$

A sufficient condition to ensure the ergodicity of the chain is that none of the conditional distributions be anywhere zero, which means that any point in the space can be reached from any other point.

A drawback of the Gibbs sampler is that if there are strong dependencies within the components $(x_1, \ldots, x_p)$, then it is likely to take a large number of iterations before it reaches convergence.

4.A.3 Sequential Importance Sampling

Sequential importance sampling (SIS) is a direct extension of Important Sampling (IS) we presented in chapter 2, and its objective is to compute expectations w.r.t. a sequence of probability measures of increasing dimension $\{\pi_i(x_{1:i}); i = 1, \ldots, n\}$ defined on $\{\mathcal{X}^i \in \mathbb{R}^i; i = 1, \ldots, n\}$ where each density is only assumed known up to a normalizing constant

$$\pi_n(x_{1:n}) = \frac{\tilde{p}_n(x_{1:n})}{Z_n^p}.$$

We assume that $\tilde{p}_n : \mathcal{X}^n \to \mathbb{R}^+$ can be evaluated pointwise but the normalizing constant Z_n^p is unknown, this in fact is the case of our target distribution (4.6).

The importance density function $q_n(x_{1:n}) = \tilde{q}_n(x_{1:n})/Z_n^q$ is defined on a support that covers the support of the target distribution π_n, and ideally captures some of the important characteristics, for example its scale or dependence structure. As

for the IS case, given $\pi(x) > 0 \Rightarrow q(x) > 0$, the Radon-Nykodym theorem allows us to write

$$\pi_n(x_{1:n}) = \frac{\omega_n(x_{1:n})\, q_n(x_{1:n})}{Z_n^p},$$

$$Z_n^p = \int \omega_n(x_{1:n})\, q_n(x_{1:n})\, dx_{1:n}$$

where the unnormalized weight function $\omega_n(x_{1:n})$ is given by the ratio

$$\omega_n(x_{1:n}) = \frac{\tilde{p}_n(x_{1:n})}{q_n(x_{1:n})}$$

Assuming the proposal density has also been conveniently chosen because it is easier to sample from, we can approximate $q_n(x_{1:n})$ using the empirical measure obtained from the N particles $(x_{1:n}^1, \ldots, x_{1:n}^N)$. Note that since the importance distribution admits the following decomposition

$$\begin{aligned} q_n(x_{1:n}) &= q_{n-1}(x_{1:n-1})\, q_n(x_n|x_{1:n-1}) \\ &= q_1(x_1) \prod_{i=2}^{n} q_i(x_i|x_{1:i-1}) \end{aligned}$$

each particle $x_{1:n}^j$ for $j = 1, \ldots, N$ has been derived recursively starting by sampling $X_1^i \sim q_1(x_1)$ at time 1 and then iterating $X_i^j \sim q_i(x_i|x_{1:i-1}^j)$ at time i for $i = 2, \ldots, n$.

Using the sampled particles we can approximate the target distribution

$$\hat{\pi}_n(x_{1:n}) = \sum_{j=1}^{N} W^j\, \delta_{X^j}(x_{1:n})$$

where the normalized weight $W_n^j = \omega_n(X_{1:n}^j) \,/\, \sum_{l=1}^{N} \omega(X_{1:n}^l)$ is derived from the unnormalized weight computed recursively

$$\omega_n(x_{1:n}) = \omega_1(x_1) \prod_{i=2}^{n} \alpha_i(x_{1:i})$$

given

$$\alpha_n(x_{1:n}) = \frac{\tilde{p}_n(x_{1:n})}{\tilde{p}_{n-1}(x_{1:n-1})\, q_n(x_n|x_{1:n-1})}$$

the incremental importance weight.

Chapter 5

Application to Gene Expression Data

5.1 Introduction

Microarray gene expressions studies are routinely carried out to measure the transcription levels of an organism's genes. A common aim in the analysis of expressions measurements observed in a population is the identification of naturally occurring sub-populations. In cancer studies, for instance, the identification of sub-groups of tumours having distinct mRNA profiles can help discover molecular fingerprints that will define subtypes of disease (Smolkin and Ghosh, 2003).

Many different approaches have been suggested for partitioning biological samples, including hierarchical clustering, K-means and probabilistic methods based on finite mixtures of distributions. One of the widely recognised advantages of model-based clustering lies in the fact that it explicitly accounts for the experimental noise that is typical of microarray studies (Liu and Rattray, 2010). The gene expression measurement within each cluster are modelled as random variables drawn from a group-specific probability distribution, several well-known parameter estimation algorithms exist and have been applied to gene expression data (Qu and Xu, 2004; He et al., 2006; Liu and Rattray, 2010; Melnykov and Maitra, 2010).

In this chapter we implement the probabilistic clustering algorithms we pro-

posed in chapter 3 to robustly model a cohort of patients diagnosed with breast cancer. The aim is to identify the informative genes and rank them by importance in order to discover data clusters that emerge only when considering the expression levels of the selected genes. Being able to identify the smaller subset of informative genes is essential not only to improve the quality of the data partitioning process, but also to aid the biological interpretation of the results.

The chapter is organised as follows. In section 5.2 we describe the microarray data we intend to investigate to find evidence of clinically relevant breast cancer subtypes. In section 5.3 we select the most appropriate model parameters and fit a penalised mixture of t distributions. We also follow a resampling procedure to propose a ranking of each genes contribution towards clustering. In section 5.4.2 we analyze the marginal distribution across the proposed clusters of some relevant clinical variables and discuss the possibly different prognosis of patients in each cluster. In section 5.5 we review our results in light of similar studies conducted on the same subject.

5.2 Breast Cancer Data

The proposed sparse mixture models were used for the analysis of a publicly available breast cancer data set consisting of $n = 128$ early-stage tumors from those collected at Nottingham City Hospital NHS Trust between 1986 and 1992 (Naderi et al., 2007; Blenkiron et al., 2007). This cohort of tumours is representative of the demographics of breast cancer, and the majority of patients were post-menopausal. Microarray data recording the expression levels of $p = 36,939$ probes are available in the ArrayExpress database (www.ebi.ac.uk/arrayexpress) under accession number E-TABM-576. In particular, the total RNA from the primary breast tumors was labelled using the Illumina TotalPrep RNA Amplification kit (Ambion) following manufacturer's instructions. 1.5 ug of biotin-labelled cRNA were used for each hybridisation on Sentrix Human-6 BeadChips v1.0 (Illumina, San Diego, CA) following manufacturer's protocol. Illumina's BeadStudio software (version 3.3.8) was used to process raw array data and output a single summarised value for each

bead type on each of the arrays.

From the medical records of each patient, we also had access to several clinical variables that we can use to independently verify the relevance of the proposed clustering.

- Age: Patient age at time of diagnosis (in years).
- Meno: Menopausal status. Premenopausal (1), Postmenopausal (2).
- Size: Invasive tumour size in cm.
- Grade: Histological grade of invasive tumour (1, 2 or 3).
- Stage: Lymph node stage. Node negative (1), 3 or less axillary nodes (2), 4 or more axillary nodes, apical node, or axillary and internal mammary node involved (3).
- NPI: Nottingham Prognostic Index = (0.2 x size (cm) + grade + stage).
- ER: Estrogene Receptor (H score) system, cut off at score of 10 or DCC cut off 10fmol/mg protein).
- Dead: Alive (0), dead from breast cancer (1), dead from other causes (2), lost to follow up (3).

5.3 Penalised t Mixture Model

Our analysis is unsupervised and we fit a penalised mixture of Student's t distributions, model (3.4) discussed in chapter 3, in order to propose a reasonable model-based clustering. Within the estimation process we execute a resampling routine which provides the means for ranking genes according to their perceived contributions to the cluster assignment.

Model Assumptions

As a preliminary check, we verify that we assume the proper parametric density function to describe the gene distribution. We fit the standard non penalised version of the Gaussian and t mixtures and compute the likelihood ratio of the two models. The resulting tests statistic of 12761.71 with a p-value of less than 0.0001 suggests that our hypothesis that genes expression levels are best described by a Student's t density function is in fact supported by the data. This conclusion is confirmed also by the `BIC` which takes into account the fact that a t mixture is a less parsimonious model.

Even when we fit the marginal density of each gene individually, as in Figure 5.1, we see that in the majority of the cases the estimated degrees of freedom parameters are very low. This evidence indicates that a more accurate fit can be achieved by assuming t components rather than Gaussian ones, which is corroborated also by the distribution of p-values from the likelihood ratio tests.

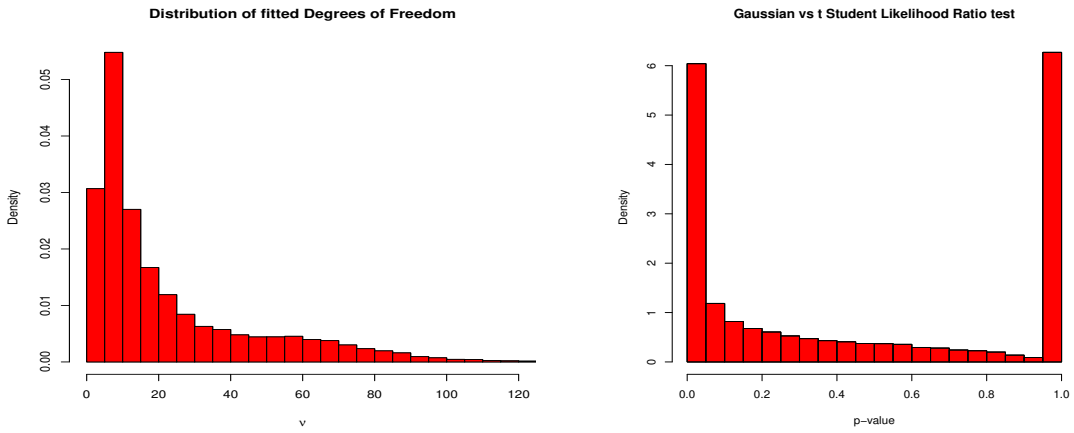

Figure 5.1: Gaussian Vs t components. Left plot shows the distribution of fitted degrees of freedom assuming a t density function. Right plot shows the distribution of $p-$values of the likelihood ratio test for all genes.

Model Selection

Following the model selection procedure detailed in section 3.4 and already applied in the financial application in section 6.6, we infer the number of clusters K by testing up to five possible components, $K = \{2, \ldots, 5\}$.

The optimal level of penalisation for the location and dispersion parameter is again found exploring all possible combinations of λ_μ and λ_σ in the interval $[0, \ldots, 10]$ and choosing the one that minimizes the modified `BIC` criterion.

In Tables 5.1, 5.2, 5.3 and 5.4 we report the detailed output of each simulation run on the grid search. For each combination of λ_μ and λ_σ we quote the distance in `BIC` units from the lowest level reached under each assumption, $K = \{2, \ldots, 5\}$. We note that despite some discontinuity caused by the different number of variables retained at each iteration, there is a progressive degradation towards the minimum point.

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	496190	299131	269765	255141	252057	250051	249673	250659	251453	252419	253433
1	304439	102158	90788	59026	85327	53969	54343	55332	76619	73764	81059
2	271544	88345	56510	27265	39927	22155	22490	42056	14582	26166	27473
3	291591	74984	42869	11451	11696	29033	10990	32777	9717	33845	31160
4	287673	56047	24521	29325	21242	19629	19365	20227	21296	27763	10760
5	259530	55970	35497	11509	22544	17473	5926	17978	6887	8242	21345
6	261046	57871	34020	12055	7397	6036	16635	17396	7581	8263	19531
7	263601	69894	26371	24081	18859	23932	7174	7704	8882	19132	20226
8	266387	62287	37557	9203	10253	4114	4212	21483	18901	22750	11035
9	267621	56702	39148	16954	20459	4521	18089	0	1309	21109	22084
10	270346	67352	24258	19126	22789	12039	3967	24917	13076	14201	15387

Table 5.1: Model Selection, grid search assuming $K = 2$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	715748	418860	368117	348721	340253	338392	363204	339281	366690	340110	344272
1	413325	160013	111277	93360	84884	108106	81609	83977	85423	93913	89178
2	370716	113910	60606	43199	35336	33235	39251	33488	36216	38076	40171
3	356324	87248	41337	22505	14832	12444	12501	14126	15972	18075	19920
4	350387	78111	64056	14979	6406	5007	4171	6361	5351	9557	12152
5	395252	80457	29918	11072	3460	10170	22965	2902	2483	6800	8622
6	363507	83024	30326	9805	3228	52	0	1872	3657	5663	7593
7	NA	83712	32418	11890	3522	65	1026	1766	3648	5626	7540
8	NA	88621	33295	13196	5095	2482	2378	2818	4646	18371	8614
9	376520	97253	36333	15379	26120	4279	4108	4970	6541	8301	10485
10	NA	102816	39925	17472	8688	5308	5304	6245	7913	9485	11943

Table 5.2: Model Selection, grid search assuming $K = 3$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

In Figure 5.2 we apply some degree of smoothing by interpolating the output

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	883033	563943	487388	459927	448919	444569	447565	445096	447074	452178	451442
1	557100	222731	156136	131376	119747	112084	113327	116789	123181	120291	122195
2	501169	154212	87895	65402	51548	50236	49368	51492	54868	55697	62570
3	486039	134467	64912	31548	22330	21550	21536	18782	24904	24922	28666
4	NA	137459	55109	26488	16483	11996	9967	14201	16216	26620	20520
5	NA	112920	48954	21375	8026	5316	4237	3773	4654	9174	13266
6	NA	135566	51272	23608	7527	3337	2046	5280	7216	6456	9097
7	NA	133335	51545	22101	9065	4632	4006	3239	8140	7295	18260
8	761167	150730	56264	23590	5992	5831	9289	2702	191	11084	8093
9	901745	149743	61223	27740	14262	6054	2071	0	729	2141	12864
10	NA	148479	65089	20873	15726	1758	1050	274	7565	7844	14558

Table 5.3: Model Selection, grid search assuming $K = 4$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	1095880	716338	625206	595298	575313	569698	562102	564070	567886	571899	572658
1	928946	289149	210275	173234	160529	151075	155095	156526	164262	156967	168482
2	NA	204744	126402	80250	69494	70533	69253	73599	80359	85251	85235
3	NA	161040	88449	53387	34103	32413	31404	33682	42835	43251	46213
4	800686	164737	65796	27065	25979	20035	23601	13683	19001	27308	29325
5	NA	169385	71684	30501	19385	12476	8191	7911	15048	12639	12243
6	1160384	213481	69246	40025	13723	8464	7156	4687	5778	15344	9124
7	865998	198420	68356	20565	3174	945	13192	8645	20034	9383	16776
8	985544	202227	63106	23342	23533	7412	0	1325	11544	8571	6752
9	NA	245857	87026	39537	14123	4604	9366	5160	5364	17736	10285
10	NA	240396	98007	44610	23186	13412	8350	8641	15357	10265	11978

Table 5.4: Model Selection, grid search assuming $K = 5$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

from previous tables and note that the optimal level of penalisation is localised in a fairly stable region even assuming different number of components.

The results summarized in Table 5.5 support a choice of a 3-components mixture. The `BIC` is at its lowest for $K = 3$ while the log likelihood tends to overfit the data by assuming too many components.

# Clusters	`LLIK`	`AIC`	`BIC`
2	-1107716	2268718	2344705
3	-1064321	2208573	2322554
4	-1034137	2174849	2326825
5	-1010667	2154553	2344524

Table 5.5: Model Selection `PTM`. Number of clusters.

5.3.1 Clustering Results

As suggested by the evidence from the experiment done with simulated data in section 3.5, we also implement a resampling step to reduce the dimension of the

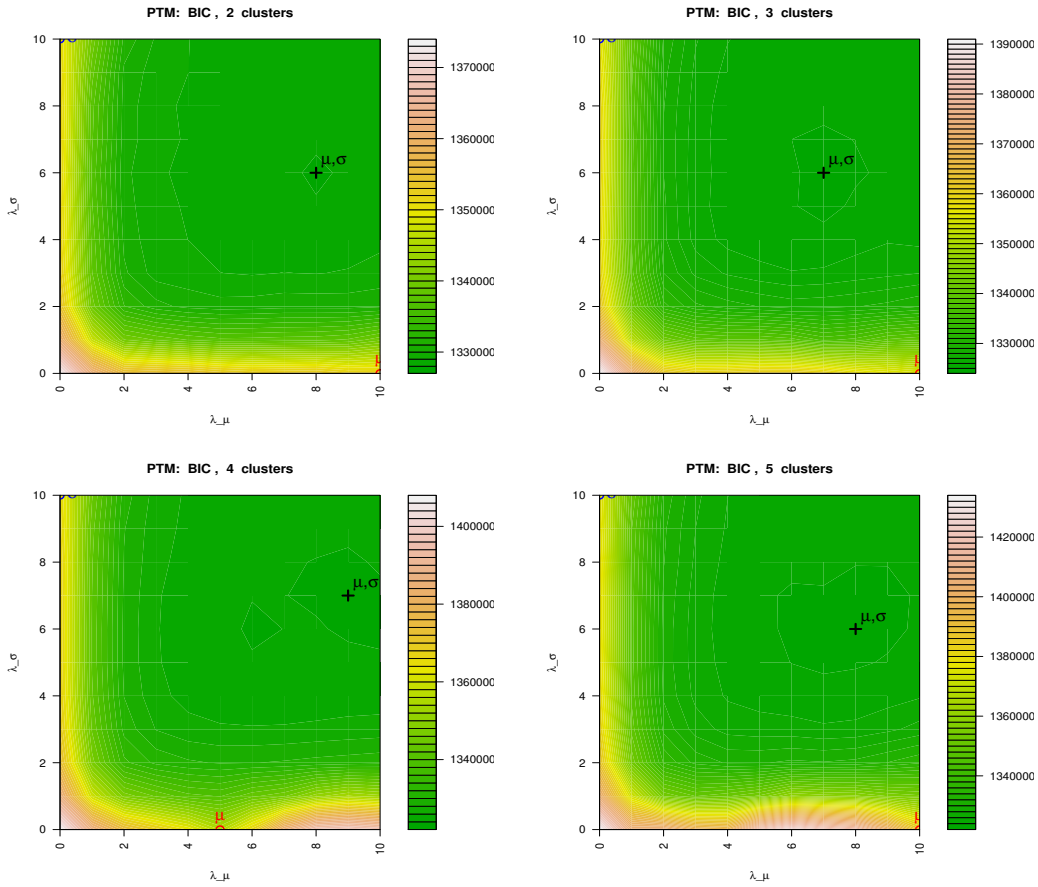

Figure 5.2: Optimal level of penalisation λ_μ and λ_σ for $K = 2, \ldots, 5$ according to modified `BIC` criterion.

data. Of the 36,939 genes recorded we only retain 1128 probes which have a selection probability above 0.7 (about 3% of the total). The results of fitting a t mixture of three components using the selected genes across the 128 tumors are displayed in Figure 5.3, where on the x-axis we have the probes and on the y-axis the patients grouped by cluster.

On first inspection, we can see that there is a noticeable correspondence between the cluster we propose and the expression level patterns of the genes depicted in the heatmap. It is clear that different genes are necessary to isolate different groups of patients. We note that, typically, the separation emerges because there is a marked under expression of certain informative genes in one cluster relative to the

other two.

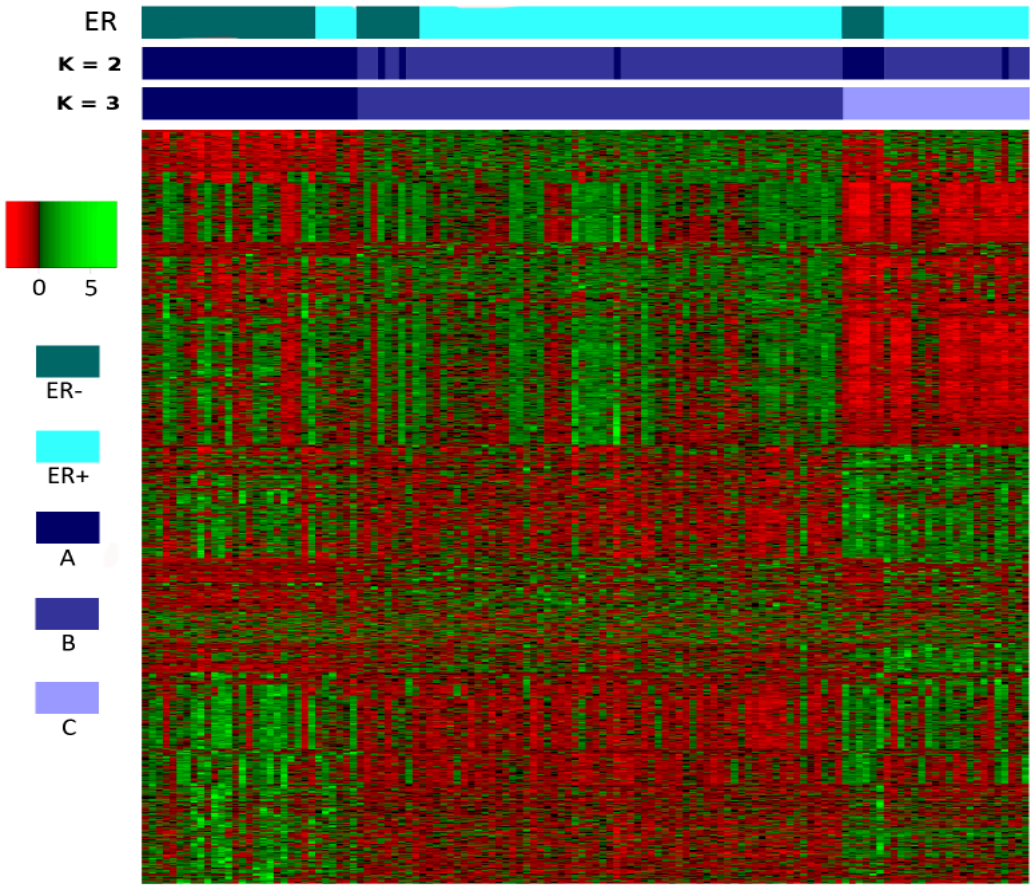

Figure 5.3: Heatmap of expression levels of selected Genes clustered assuming three component. On the side the distribution of the Estrogen Receptor (ER) factor in each cluster, and the clustering assignment assuming only two components.

In order to interpret the clustering allocations, we explore the correlation with the clinical sub-groups induced by the Estrogen Receptor (ER). The ER status as well as the patients cluster assignments are reported at the top row of Figure 5.3. Approximately 80% of human breast carcinomas present an estrogen receptor α-positive (ER+) disease, with ER+ breast cancers responding well to therapies, and ER-negative tumours being more resistant. ER status is an essential determinant of clinical and biological behaviour of human breast cancers, and it is well known that the major molecular features of breast cancer segregate differently according to ER status (Schneider et al., 2006). With $K = 2$, our analysis finds clusters that

overlap with ER-positive and ER-negative tumours, which is in line with previous findings on independent data sets (Van't Veer et al., 2002, for instance). In this case, 82.5% of all ER- tumors fall into cluster A, and the remaining ones into cluster B. When an additional cluster is added, we observe a split of the ER+ dominated cluster B into two groups: 81% of the samples in cluster A are still ER-, whereas the majority of samples in B and C are ER+ (88% and 76%, respectively).

5.3.2 Variable Selection Results

The other important property of the model we propose is that while identifying and fitting each cluster density function, it also selects the most informative variables. To verify that this is in fact the case we perform a non parametric Kruskal-Wallis rank sum test (Hollander and Wolfe, 1999) under the null hypothesis that the three clusters have been sampled from the same distribution. In Figure 5.4 we show a separate boxplot for the p-values of the selected genes and for the excluded ones. We see that for the genes that we considered informative, the null hypothesis is refused in almost all cases, whilst for the excluded genes the clustering assignment is irrelevant. On the basis of this evidence we conclude that the proposed model really identifies clusters of patients with significantly different gene expression levels and also selects the more informative genes.

One other assumption we want to see confirmed is that the selected genes have marginal Student's t distributions. In Figure 5.5 we plot the histogram of the fitted degrees of freedom parameter ν for every component. It appears that in the majority of cases the estimated density has longer tails than a Gaussian, therefore it justifies assuming low degrees of freedom t density as suggested also by the distribution of the p-values of the likelihood ratio test.

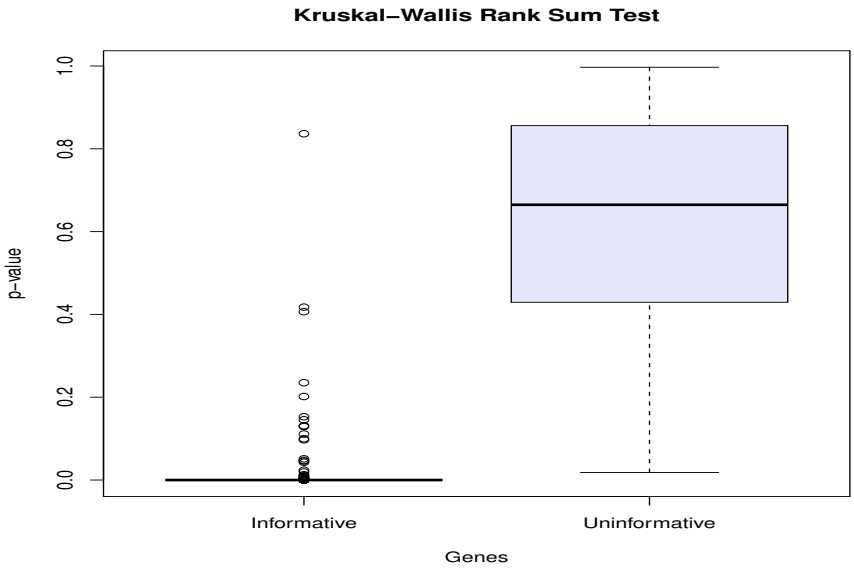

Figure 5.4: Kruskal-Wallis rank sum test under the null hypothesis that the three clusters identified come from the same distribution. Left boxplot represents the distribution of p-values of the test conducted on the selected genes. Right boxplot is the same test on the excluded variables.

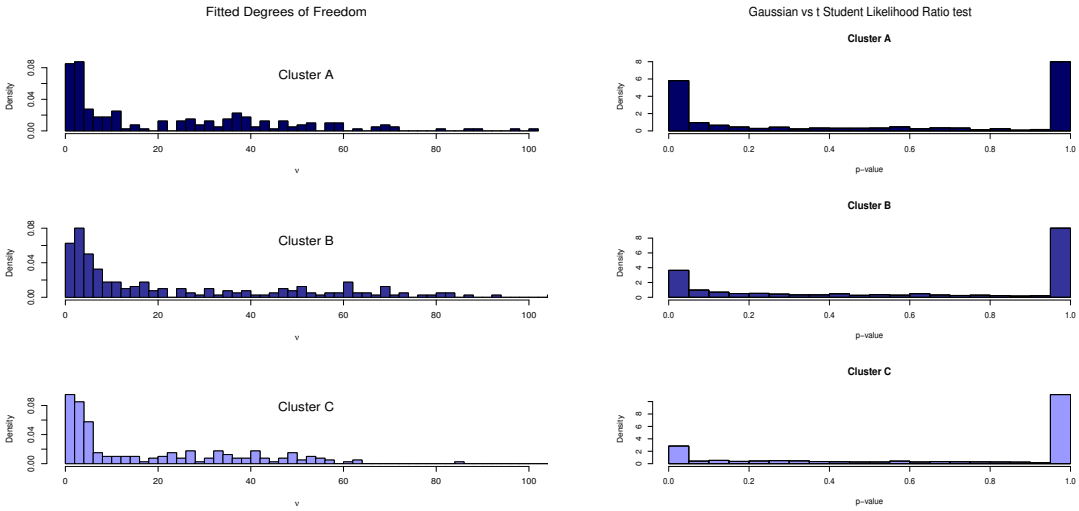

Figure 5.5: Distribution of degrees of freedom parameter ν fitted on the selected genes by cluster. Right plot shows the distribution of $p-$values of the likelihood ratio test by cluster.

5.4 Biological Explanation

5.4.1 Gene Ranking

Several of the genes retained by the model since their selection probability is above the threshold 0.7, had also been previously included in independent ER signatures (Abba et al., 2005; Thakkar et al., 2010). Among the highest ranking genes we find (with selection probabilities): FOXA1 (0.935), NTN4 (0.906), CSNK1A1 (0.892), STC2 (0.885), SF3B3 (0.863), MMP12 (0.860), ITGB7 (0.845), CA12 (0.831), MLPH (0.827), GATA3 (0.824), NAT1 (0.802), GABRP (0.781), DUSP4 (0.745), NUTF2 (0.777), XBP1 (0.756), SLC43A3 (0.752), PLAT (0.727), ESR1 (0.727) and AARS (0.713). For example, FOXA1 has very recently been found to be a key determinant of ER function and endocrine response (Hurtado et al., 2011). Moreover, FOXA1, GATA3 and ESR1 have also been found to be associated to ER status in an analysis of invasive ductal carcinoma (Schneider et al., 2006).

In Figure 5.6 we plot the fitted component density functions of FOXA1, GATA3, ESR1 and NAT1. The expression level of these genes for patients in cluster A follow a clearly separate and distinct distribution from the other two clusters which are ER+ dominated. To appreciate how important this separation can be, note that genetic variants immediately upstream of ESR1 have been linked to breast cancer risk.

Very recently, A.K. Dunbier, H. Anderson, Z. Ghazoui, E. Lopez-Knowles, S. Pancholi, R. Ribas, S. Drury, K. Sidhu, A. Leary, L. Martin (2011) found that three open reading frames within this region are tightly co-expressed with ESR1, and investigated the function of these three genes: C6ORF97, C6ORF96 and C6ORF211. Their findings suggest that the genes could contribute to the phenotype associated with ER positivity. In addition, they may be involved in the mechanism by which genetic variation in this region of the genome contributes to breast cancer susceptibility. These three genes have also been found to have high selection probabilities in our ranking (0.770, 0.713 and 0.709, respectively).

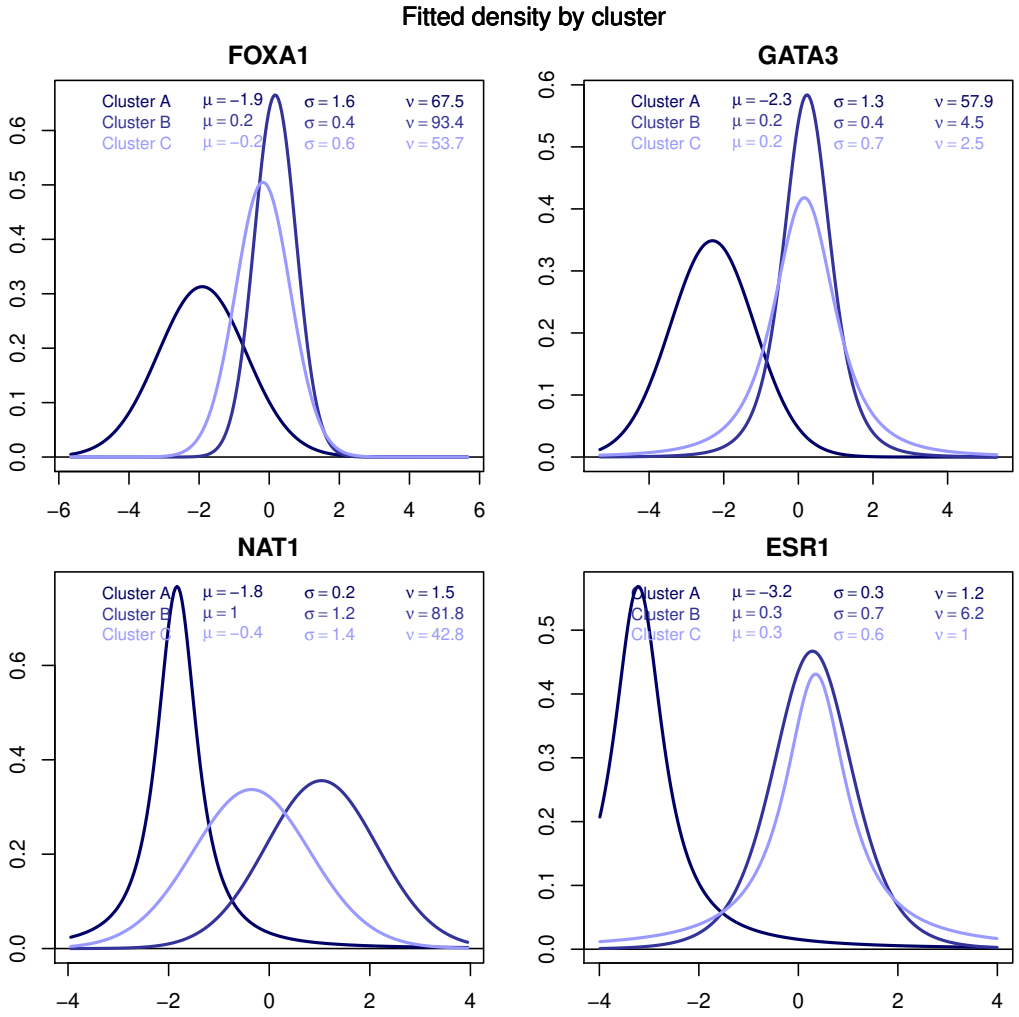

Figure 5.6: ER related genes selected by resampling methods.

5.4.2 Clinical Variables by Cluster

To further investigate whether the three clusters may indicate clinically distinct subgroups of patients, we explore the frequency distribution of the NPI, grade, tumor size and survival rate. In this Section we report on the prognosis profile of the 128 patients, broken down by cluster. When using two mixture components, all the variables display a markedly different distribution in the two clusters, which are representative of ER+ and ER- tumors. When using three mixture components, all variables are differently distributed in the ER+ dominated clusters B and C,

with cluster C being more similar to cluster A, which is ER- enriched.

Nottingham Prognostic Index

The NPI is an important index used to determine prognosis following surgery for breast cancer. This index uses pathological variables, such as nodal status, tumor size and histological grade, to generate a prognostic score for each patient that is predictive of outcome (Callagy et al., 2006). Since it is a continuous variable, NPI offers a responsive and sensitive means of modelling a continuum of clinical aggressiveness, indexing the outcome likelihood of invasive breast cancer patients.

In Figure 5.7 we report the boxplot of the NPI marginal distribution across different clusters of patients. We find that cluster A, which presents mainly ER- tumors, is characterized by an higher level NPI. Interestingly in the $K = 3$ case, we see a split of the ER+ dominated cohort, where cluster B shows a more benign NPI profile. We can verify that the differences we observe are statistically significantly by computing the Kolmogorov-Smirnov test of equality. The p-values of the paired test, $A/B = 0.0005$, $A/C = 0.9517$, $B/C = 0.0090$, confirm that, in terms of NPI profile, cluster A and C are much more similar despite having different ER status.

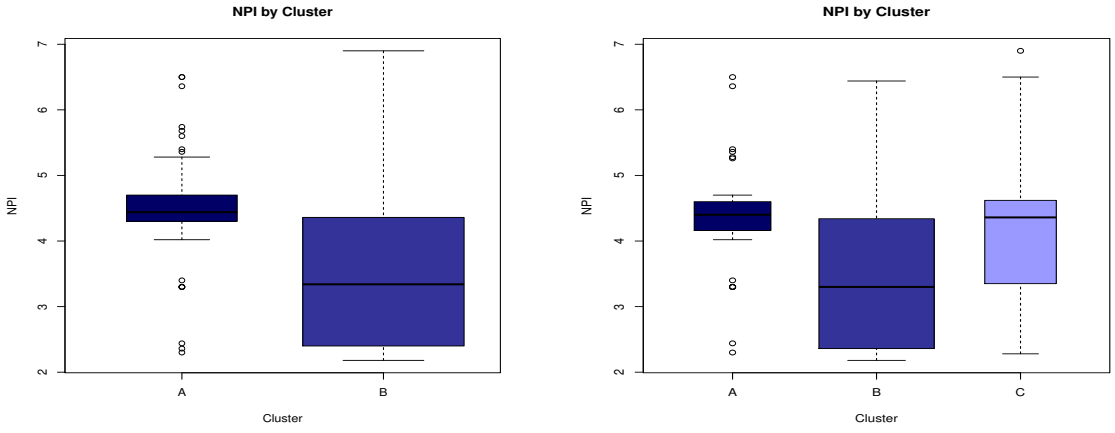

Figure 5.7: NPI distribution by cluster assuming two and three components.

Another important test that support this conclusion is the χ^2 test for independence. Since NPI is a continuous variable, we resolve to bin the observations in

quartiles in order to compute the contingency table 5.6. We find that the p-value of the χ^2 test is 0 for $K = 2$ and 0.003 for $K = 3$ which reflects the significant deviation from what would be a random cluster assignment.

	K=2		K=3			
Quartile	Cluster A	Cluster B	Cluster A	Cluster B	Cluster C	Expected
1	0.07	0.33	0.06	0.36	0.15	0.24
2	0.15	0.30	0.19	0.30	0.19	0.25
3	0.39	0.23	0.39	0.20	0.37	0.28
4	0.39	0.14	0.35	0.13	0.30	0.22

Table 5.6: NPI contingency table assuming two and three components.

Grade

One other clinical variable that appears to have significantly different distribution across the clusters we propose is Grade. It is a measure of cell appearance, where higher grade means an higher cell life alteration caused by the tumor.

In Figure 5.8 we find a similar pattern to the one we encountered for NPI. Cluster A is generally showing a less favourable prognosis which, in the $K = 3$ case, is close to the Cluster C. As before, this impression is confirmed by the p-values of the Kolmogorov-Smirnov test of equality: $A/B = 0.0$, $A/C = 0.9998$, $B/C = 0.0010$.

In the contingency table 5.7 we report the exact frequency of each state in each cluster. The p-value of the χ^2 test for independence is 0 for $K = 2$ and 0 for $K = 3$ which would suggest that the partition we fitted is not random.

	K=2		K=3			
Grade	Cluster A	Cluster B	Cluster A	Cluster B	Cluster C	Expected
1	0.07	0.34	0.06	0.39	0.15	0.26
2	0.29	0.46	0.29	0.50	0.30	0.41
3	0.63	0.20	0.65	0.11	0.56	0.34

Table 5.7: Grade contingency table, assuming two and three components.

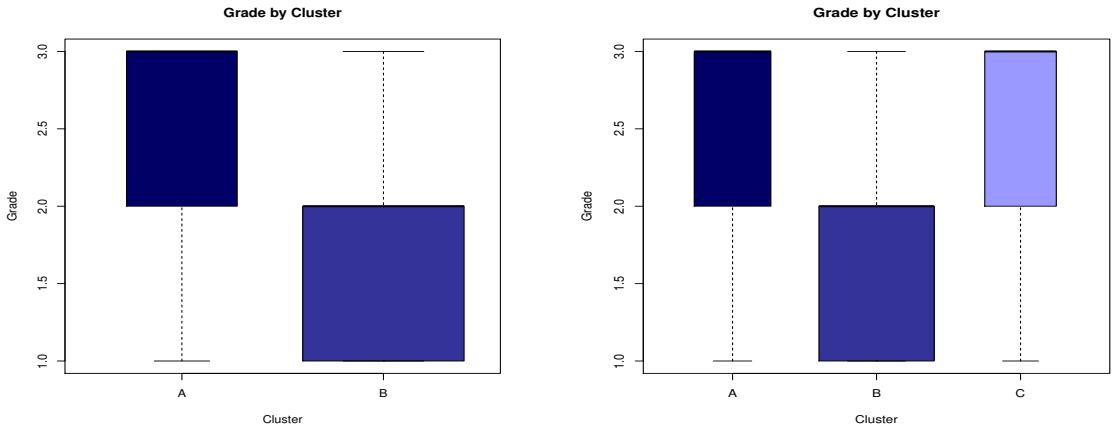

Figure 5.8: Grade of invasive tumour by cluster assuming two and three components.

Size

The size of invasive tumor seems to be another obvious clinical variable we should monitor to asses the relevance of our clustering results. In Figure 5.9 and Table 5.8 we report the boxplots and frequency table. In this case the p-values of the Kolmogorov-Smirnov test of equality, $A/B = 0.2273$, $A/C = 0.5387$, $B/C = 0.0022$, seem to suggest that only Cluster B and C are significantly different. Similarly only the p-value of the χ^2 test for $K = 3$ is significant 0.008, whereas for $K = 2$ is 0.17.

	K=2		K=3			
Size	Cluster A	Cluster B	Cluster A	Cluster B	Cluster C	Expected
1	0.25	0.45	0.30	0.53	0.11	0.39
2	0.10	0.11	0.10	0.09	0.19	0.11
3	0.28	0.30	0.30	0.26	0.37	0.29
4	0.38	0.14	0.30	0.13	0.33	0.21

Table 5.8: Size of invasive tumour (in cm) by cluster assuming two and three components.

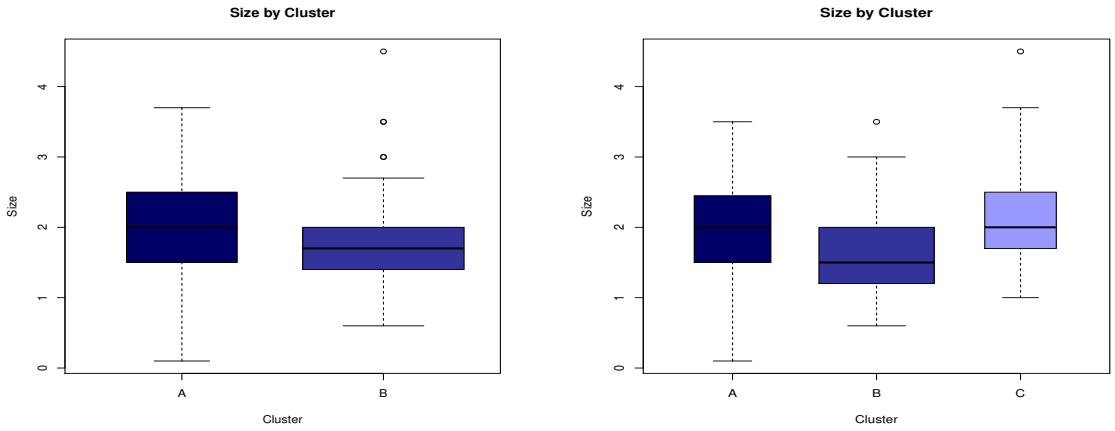

Figure 5.9: Size contingency table assuming two and three components.

Observed Survival Rate

Ultimately, one of the most important factors to consider is the survival rate of the patients within each cluster. For this analysis we only look at the patients for which we have follow up information at the time when the data were collected. In this case we only have a smaller subset of 115 patients excluding those for which we do not have after surgery information and those that have died for other reasons. In Figure 5.10 we plot the ratio of the patients whose medical record indicate they have not died by cancer after the surgery and the subsequent treatment.

At first inspection, the pattern we observe seem to be in line with the profile of the other clinical variables, where cluster B has an higher expectation of survival. The Kolmogorov-Smirnov test on paired clusters returned the following p-values: $A/B = 0.6803$, $A/C = 0.9994$, $B/C = 0.1766$, which suggests that the only significant difference is between cluster B and cluster C, which otherwise shows a survival rate fairly similar to cluster A.

The χ^2 test based on the contingency table 5.9 considers the marginal distribution of all clusters simultaneously and provides a stronger evidence that they are not independent since the p-value 0.035 is significant for $K = 3$. On the other hand, with a p-value 0.219 for $K = 2$ we can not affirm that the survival rate of the patients is significantly different under the two clusters assumption.

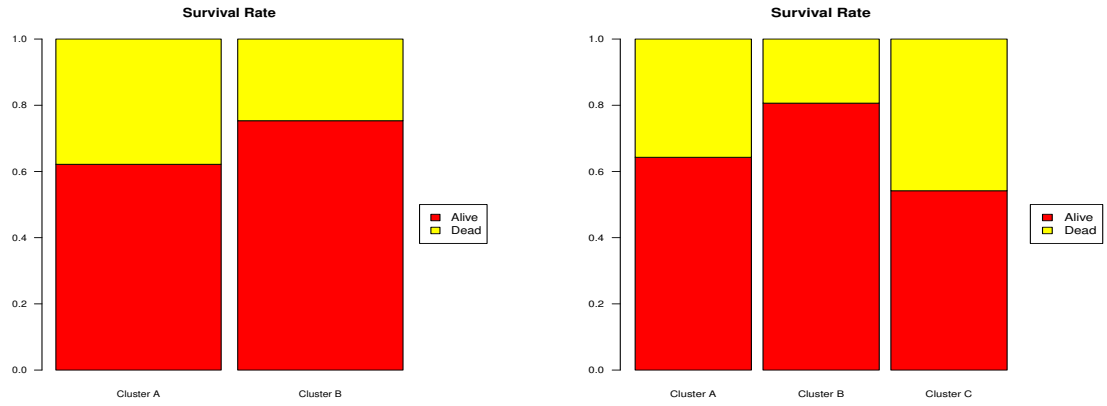

Figure 5.10: Observed Survival rate by cluster assuming two and three components.

	K=2		K=3			
	Cluster A	Cluster B	Cluster A	Cluster B	Cluster C	Expected
Alive	0.62	0.75	0.64	0.81	0.54	0.71
Dead	0.38	0.25	0.36	0.19	0.46	0.29

Table 5.9: Observed Survival rate by cluster assuming two and three components.

5.5 Validation with other Studies

Different studies have tried to identify breast carcinomas subtypes beyond the simple classification between positive and negative Estrogen Receptor status (Sorlie et al., 2001; Calza et al., 2006; Sotiriou and Pusztai, 2009; Nicolau et al., 2011). The number of subtypes is not consistent across papers and there is no clear consensus on the gene expression signatures that should mark each group.

In order to validate our results, we have compared our proposed clustering assignment with the clustering obtained by fitting a t mixture on the list of genes identified as relevant by Calza et al. (2006). The 87 genes used by Calza et al. (2006) is a subset of the 500 genes that were originally selected by Sorlie et al. (2003) who isolated those that showed the most informative variation among subtypes of tumors. In their study of 412 patients from Stockholm and Uppsala,

Sweden, genes were median-centered and subtypes were isolated using an average-linkage hierarchical clustering algorithm based on uncentered correlation distance metric.

The heatmap in Figure 5.11 shows the gene expression patterns in our cohort of 128 samples using the *Basal-like* signatures provided by Calza et al. (2006). Comparing their cluster assignment with ours we find that the *Basal-like* subtype can be positively mapped to our cluster A. Not only is it characterised as expected by an higher percentage of ER- cases, but we observe a consistent over-expression of the relevant genes in correspondence of Cluster A. The clinical prognosis also is in line with what is described by previous papers: patients of this group are expected to have an above average tumor size, see Figure 5.9, an higher frequency of grade three carcinomas, Figure 5.8, and they are also less responsive to therapy, Figure 5.10.

Within the ER+ dominated subgroup, there is evidence to suggest a correspondence between our cluster B and the Luminal A subtype, whereas our cluster C seems to correspond to the Luminal B subtype. Previous studies have generally found it difficult to separate the two Luminal subtypes and suggested different lists of marker genes to isolate them. We again apply the signature used by Calza et al. (2006) to our cohort of 128 tumor, and the resulting heatmap is plotted in Figure 5.12.

We note a clear separation between our cluster A and the remaining two. Although the separation between cluster B and C is less clear, we can justify the three clusters given that the genes in cluster B are generally the most over-expressed. Figure 5.13 shows the expression profile of the Luminal B genes. In this case too we can identify three main patterns overlapping with our three clusters; notably, the gene expression patterns in cluster C appear to be more similar to cluster A, despite their different ER status. This result has also been confirmed by other studies in which Luminal B and ER- dominated subtypes have been found to share similar grade-associated and outcome-associated genes expression levels (Sorlie et al., 2003; Calza et al., 2006). The differences in prognosis between cluster B and C can also be seen in Figures 5.9, 5.8, and are in line with the results described by Sorlie

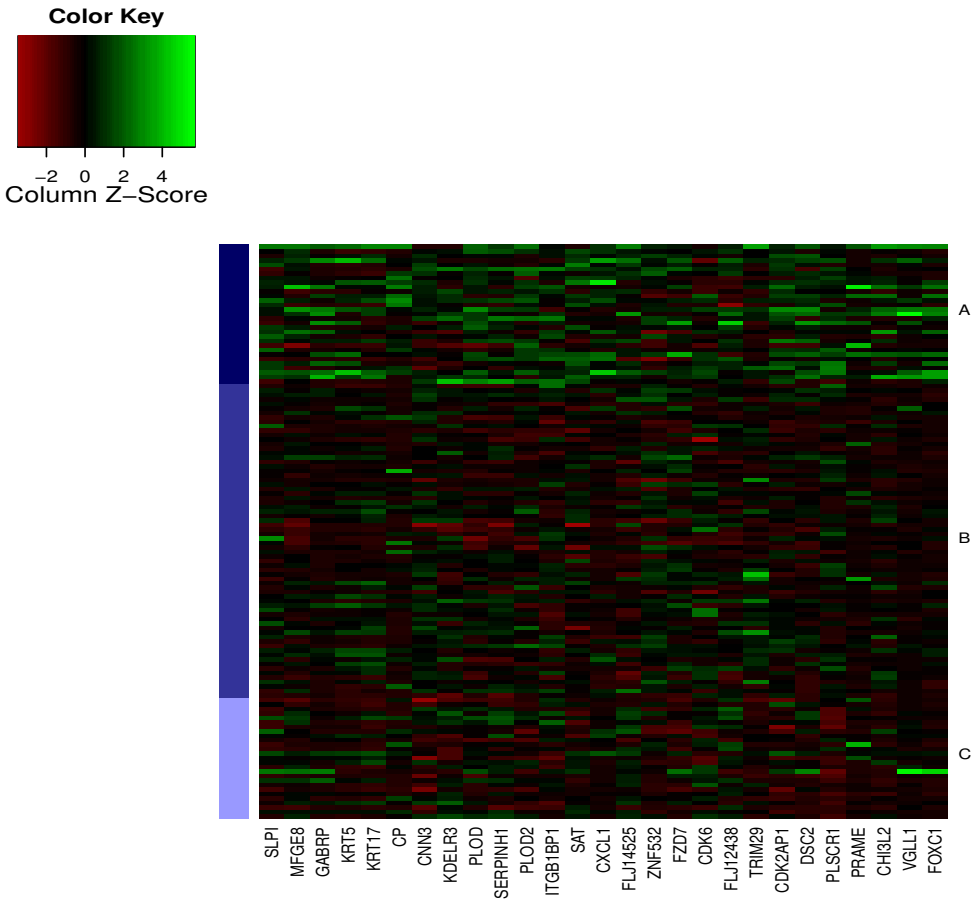

Figure 5.11: *Basal-like* markers genes

et al. (2001) and Nicolau et al. (2011) for Luminal A and B subtypes. Luminal B patients (as in our cluster C) present a bigger tumor size, higher frequency of grade 3 carcinomas and ultimately a less favourable survival rate. These properties make them more similar to *Basal-like* subtype (as in our cluster A), than to Luminal A (as in our cluster B).

In order to make a direct comparison between the cluster assignments we would have obtained had we used the signature of Calza et al. (2006), and the cluster assignment we propose, we have fitted a (non-penalised) three-component mixture model using the 87 genes of Calza et al. (2006). Table 5.10 summarises the outcome of this comparison. We find that there is a fair agreement and that most of the patients are assigned to the same clusters. Cluster A, that is the ER- dominated

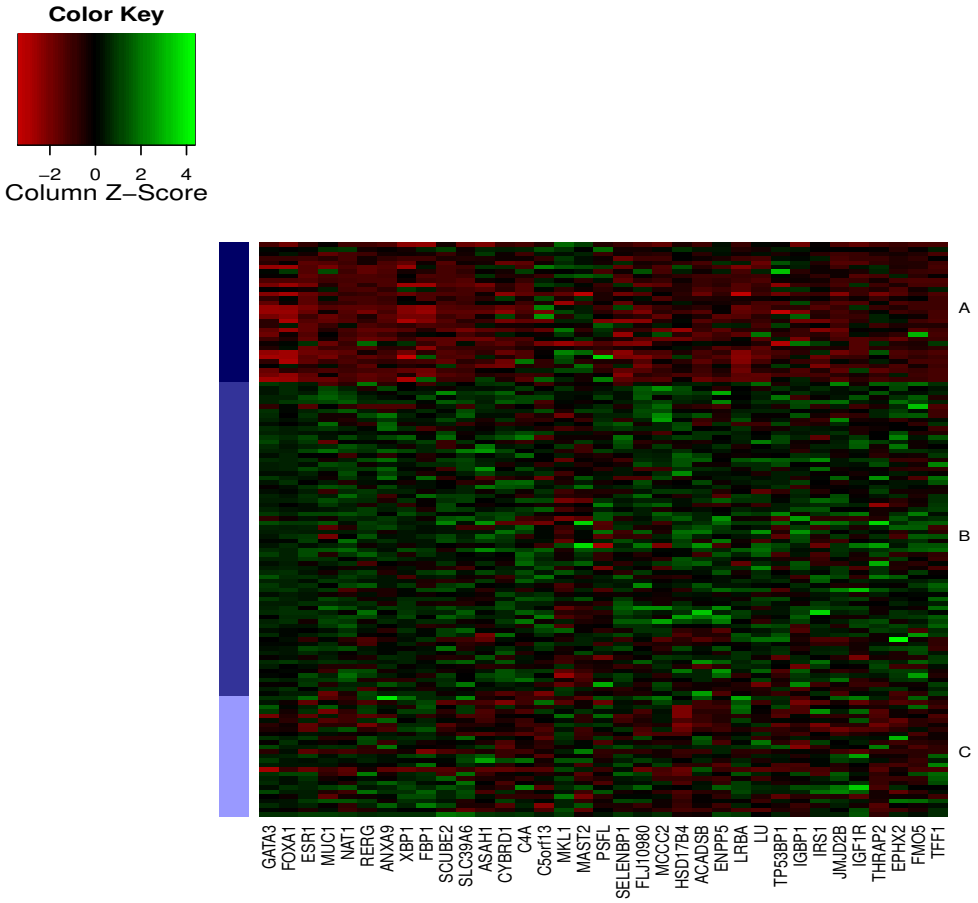

Figure 5.12: Luminal A genes list used in the study of the Uppsala cohort

group of patients that we mapped onto *Basal-like* subtype, is the one which is more easily identifiable and it shows an almost exact match between the two models. On the other hand, there is slightly more disagreement when classifying the ER+ patients. Although the overlap between models is still clear, as noted before, the separation of Luminal A and Luminal B subtypes is usually more arduous.

5.6 Discussion

In this chapter we have applied the penalised mixture of Student's t distribution we developed in chapter 3 to model a cohort of 128 patients diagnosed with breast cancer. A resampling procedure for model selection and variable ranking has also

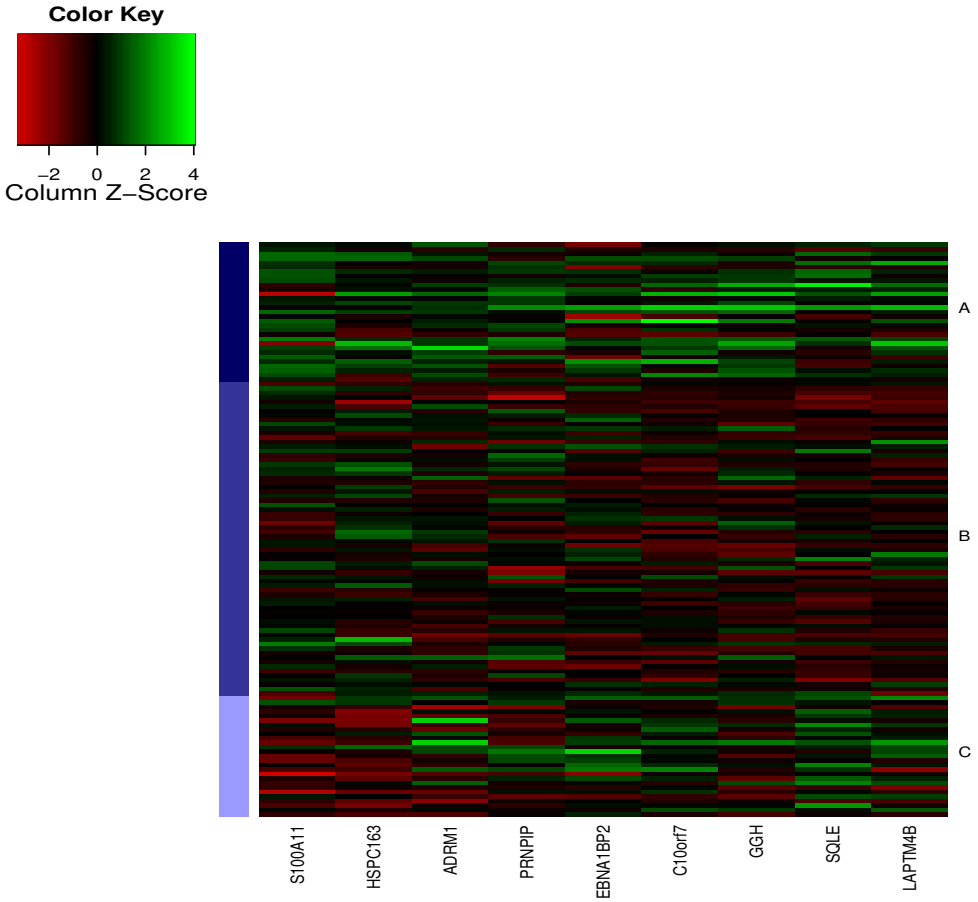

Figure 5.13: Luminal B genes

		PTM		
		Cluster A	Cluster B	Cluster C
TM	Cluster A	27	0	3
	Cluster B	1	58	3
	Cluster C	3	12	21

Table 5.10: Overlap between the cluster assignment we propose, PTM, using 1128 genes, and the cluster assignment obtained fitting a t-mixture model on the 87 genes identified by Calza et al. (2006)

been implemented, which allowed us to reduce the number of genes to fit to 3% of the original data dimensions.

We identify three clinically important subtypes that are characterized by distinct gene expression profiles. We find that our proposed clustering is able to

clearly separate the ER+ from ER- samples and independently confirm the results of other studies on this subject. Moreover, there is an indication that the selected genes may be important to explain the differences regarding the marginal distribution of certain relevant clinical variables and the typical medical prognosis we observe in each cluster.

Further investigation might try to directly regress the clinical profile of each patient on the genes we have identified as informative. Positive results in this area would improve our knowledge of the relation between over/under expression of certain genes and the insurgency of specific tumor subtype.

Chapter 6

Application to Financial Data

6.1 Introduction

In chapters 3 and 4, we have presented two different methods for clustering heterogeneous data and selecting informative variables. The first method follows an unsupervised approach and consist in fitting a mixture of Student's t distributions using an EM algorithm. The second is a mixture of Lasso regressions with t errors, fitted using a PMCMC simulation procedure. We now want to adopt both models to propose a reasonable clustering of financial markets, based on a subset of measurable features of their price dynamics. The ultimate goal is to find a more appropriate systematic trading strategy whose parameters can be robustly calibrated on groups of similar markets.

In order to produce a reliable clustering, we first need to identify the relevant features that characterize each financial market. Striving to be as objective as possible we rely only on the observed price history from which we try to infer the properties of the underlying market dynamics. We derive the distribution and time series of returns and discuss what are the most appropriate statistics that we can use to measure their important features.

Once we have computed all the relevant statistics for each market, we collect them in data matrix that we then use as basis for the clustering algorithm. We fit both of our models and discuss what the informative statistics are, that have

driven the clustering and whether the commonly accepted partition based on macro sectors is justified by the evidence we found.

The outline of the chapter is as follows. In section 6.2 we present the list of financial markets we intend to cluster and describe how to preprocess historical price data to obtain a more stationary series of returns. In section 6.3 we discuss what the relevant features of the returns distributions are and the related statistics that can be used to measure them. In section 6.4 we analyze the time series of returns and focus our attention on the statistics that measure the temporal dependence and scaling properties of returns. In section 6.5 we collect all statistics in a data matrix that we are going to fit using our proposed models. We also compute an index that represents the investment performance of a simple systematic trading strategy and take it to be the response variable that we would like to cluster. In section 6.6 and 6.7 we review the clustering and variable selection results of the penalised t mixture model and Lasso regression model respectively. We illustrate the evidence in support of the conclusions we draw and find it to be consistent with our knowledge of the problem. In appendix 6.A we provide some background notions on the theoretical distributions and random process that are commonly adopted in finance to fit the distributions and time series of returns.

6.2 Data

The dataset analysed has been kindly provided by AHL Research, a quantitative investment manager, and integrates external sources with proprietary records of live prices sampled during actual trading activity. The selection of markets considered covers several sectors, assets classes and regions. The details of each market considered are listed in Table 6.1. The frequency of the samples is daily, typically the end of day official settlement price, whenever the exchange provides one.

6.2.1 Macro Sectors

Financial markets are commonly grouped in seven macro sectors based on the *fundamental* nature of the underlying good traded in that market. Here we give a

MARKET	SECTOR	DESCRIPTION	EXCHANGE	TYPE	START	CCY
ADL	METALS	Aluminium	LME	C	19900101	US
CPN	METALS	Copper.NY	COMEX	F	19900101	US
GLN	METALS	Gold	COMEX	F	19900101	US
SLN	METALS	Silver	COMEX	F	19900101	US
CRC	AGS	Corn	CBOT	F	19900101	US
WHC	AGS	Wheat	CBOT	F	19900101	US
SBC	AGS	Soyabeans	CBOT	F	19900101	US
SGN	AGS	Sugar	CSCE	F	19900101	US
CFN	AGS	Coffee.NY	CSCE	F	19900101	US
CCN	AGS	Cocoa.NY	CSCE	F	19900101	US
CTN	AGS	NY.Cotton	NYCE	F	19900101	US
EUUS	CURRENCY	Euro Vs USD	IB	X	19900101	US
ADUS	CURRENCY	Australian D Vs USD	IB	X	19900101	US
SFUS	CURRENCY	Swiss Franc Vs USD	IB	X	19900101	US
UKUS	CURRENCY	British Pound Vs USD	IB	X	19900101	US
YNUS	CURRENCY	Japanese Yen Vs USD	IB	X	19900101	US
EUYN	CURRENCY	Japanese Yen Vs Euro	IB	X	19900101	YN
CDUS	CURRENCY	Canadian Dollar Vs USD	IB	X	19900101	US
ESPC	STOCKS	E.mini.SP500.Future	CME	F	19900101	US
TSM	STOCKS	SP.Canada.60.Ind	MON	F	19900101	CD
FTL	STOCKS	FTSE	LIFFE	F	19900101	UK
DXF	STOCKS	Dax.Index	DTB	F	19901123	EU
CGP	STOCKS	CAC.40.10EUR	MATIF	F	19900101	EU
NKS	STOCKS	Nikkei.225	SIMEX	F	19900101	YN
TSJ	STOCKS	Tokyo.Stk.Exch	TSE	F	19920722	YN
AOSS	STOCKS	Ausi.SPI200.Ind	SFE	F	19900101	AD
ESTF	STOCKS	Euro.STOXX	DTB	F	20000609	EU
HSH	STOCKS	Hang.Seng	HKFE	F	19900101	HK
KIS	STOCKS	Korean.KOSPI200.Ind	KSE	F	20000920	KW
TWS	STOCKS	Taiwan.MSCI.Ind	SIMEX	F	19970109	US
TNC	BONDS	10yr.T.Notes	CBOT	F	19900101	US
GTL	BONDS	Gilts	LIFFE	F	19900101	UK
DBF	BONDS	Euro.BUND	EUREX	F	19900101	EU
JBT	BONDS	Japanese.Bond	TSE	F	19900101	YN
ABS	BONDS	Ausi.10yr.Bond	SFE	F	19900101	AD
CBM	BONDS	Canadian.Bond	MON	F	19900214	CD
EDC	IRATES	Eurodollar	CME	F	19900101	US
SSL	IRATES	Short.Sterling	LIFFE	F	19900101	UK
EUL	IRATES	Euribor	LIFFE	F	19900101	EU
EYT	IRATES	Euroyen	TIFFE	F	19900101	YN
ARS	IRATES	Ausi.T.Bills	SFE	F	19900101	AD
NGN	ENERGY	Natural.Gas	NYMEX	F	19900403	US
CLN	ENERGY	Crude.Oil.NY	NYMEX	F	19900101	US
HON	ENERGY	Heating.Oil	NYMEX	F	19900101	US
RBN	ENERGY	RBOB.Gasoline	NYMEX	F	19900101	US
PTL	ENERGY	Gas.Oil	IPE	F	19900101	US

Table 6.1: List of financial markets and macro sectors considered. Legend: **Market:** Three letters code to identify each market. **Sector:** Each market belongs to one of the seven macro sector. **Description:** Short description of the market. **Exchange:** Main exchange where the instrument is traded. **Type:** The type of contract used to execute the transaction. It can be Cash, `X`, if the good is traded on the spot, like currencies, exchange traded futures, `F` or less standardised forwards contracts `C`. **Start:** The first date the daily records are available from.**CCY:** Currency in which contract is denominated.

brief description of the major characteristics of each sector.

Metals: This sector includes all markets trading the raw goods used mainly in heavy industry. Historically, being dominated by specialist agents, it has not been very liquid nor transparent. It typically operates over the counter with a closed group of dedicated market makers. Another peculiarity is that the production can only respond slowly to surge in demand. The reason is that to increase production typically it is necessary to make a considerable investment of money and time, e.g. opening new mines. The consequence of this inelasticity is protracted imbalances between demand and offer and wide fluctuations in prices.

Stocks: This sector includes all the markets where shares of ownership of Public Companies are traded. It is the biggest in terms of capital invested and the one that attracts most attention. It is recognised as one of the most efficient since any imbalances between supply and demand are quickly absorbed and any deviation is promptly corrected. Its liquidity and transparency make it one of the most difficult sectors to predict.

Bonds: This sector contains all markets dealing with sovereign government and corporate debt obligations. It is commonly perceived to be a risk free investment. Being the safest of all alternatives available it is also the benchmark for evaluating the risk reward profile of any other asset class. It is mainly driven by countries' fiscal policies and central banks' monetary interventions.

Short Term Interest rates: This sector covers inter-banks and government short term loans markets. Similarly to the bond sector, it responds to central bank monetary policies and it is strictly linked to business cycles. It is also a measure of the short term cost of borrowing. Any fluctuation in this sector due to interest rates movements has direct repercussion throughout all other assets classes.

Energies: This sector includes all markets dealing with fossil fuels and other goods that can be used as a source of energy. It shows a seasonal pattern due to the different consumption level during warm and cold weather. It is also linked to the economy long term cycle since a booming economy does require an higher usage of energy. Historically it has also shown abnormally high volatility due to its strategic and geopolitical importance.

Agriculturals: This sector includes those markets trading the raw output of the agricultural industry. It is dominated by producers of the goods, consumers and specialist market makers. Being linked to farms cycles it has an high degree of seasonality. Crops are also very sensitive to the weather conditions during particular phases of their growth. For this reason, meteorological events can influence the price patterns of these markets. Note that producers are systematically interested in hedging future harvest, e.g. selling next year's crop to buy the seeds to plant today. Conversely, the industries that process these goods want to have guaranteed delivery of the raw material to their factories. Market makers provide the extra liquidity to both parties.

Currencies: This sector includes those markets that exchange liquid assets and liabilities which are denominated in different currencies. Their operativity is important to set the relative value of one currency versus another and has an impact on the flow of goods and capital between countries. It ultimately effects all other sectors since every financial contract in any markets has to be denominated in a particular currency. The currency markets are not centralised in any physical exchange and operate virtually for 24 hours a day.

Market's Returns

Financial markets can be thought as complex dynamical systems whose evolution drives the price or volume processes. The price change in response to incoming news or to demand vs offer imbalances, is the most informative and closely watched projection of market dynamics. In the book by Mantegna and Stanley (1999), three different methods have been suggested to compute returns series from price series.

If p is the price of an exchange traded instrument at time t and δ is the time interval, we obtain:

$$y_{t,\delta} = p_t - p_{t-\delta}.$$

Note that for daily returns we will omit the index $\delta = 1$, and use the simpler notation $y_t = p_t - p_{t-1}$. The price difference maintains the historical accuracy of the monetary variations, e.g. the daily price excursion necessary to compute historical

profits and losses. On the other hand, the magnitude of price changes is usually proportional to the nominal level of prices. In this case, it would be inappropriate to compare returns of two periods when prices were orders of magnitude apart. The volatility measure, for example, would be biased towards the time of high nominal prices. The percentage change in prices is computed as

$$y'_{t,\delta} = \frac{p_t - p_{t-\delta}}{p_{t-\delta}}.$$

It partially corrects the distortion on the volatility measure but it is still affected by changes in scale. Alternatively, the changes in natural logarithm of prices

$$y''_{t,\delta} = \log p_t - \log p_{t-\delta}$$

should correct for the average drift in scale, but it does that by introducing a non linear transformation. The second order effects in this case are difficult to assess.

Normalisation

One known problem of financial time series is that they are not stationary. The economy is an open system that evolves through time. To make historical returns more homogeneous and comparable we adopt the solution of normalising the series of returns by a rolling volatility measure, as shown in Figure 6.1. This transformation is intended also to filter out the bias introduced by the short memory of the volatility process, see Figure 6.2.

We now take the opportunity to clarify some notations used in the remainder of the chapter: $\boldsymbol{y}$ denotes the collection of returns computed as price differences; $(y_t)_{t=1}^T$ is the series of daily returns that preserves the magnitude and the sign of historical prices excursions; $\boldsymbol{y}^{(v)}$ denotes the collection of normalised returns. They are computed by dividing the price difference by a lagged rolling volatility measure

$$y_t^{(v)} = \frac{y_t}{\texttt{vol}_{t-1}} \tag{6.1}$$

where `vol` is computed as exponentially weighted moving average of the absolute price differences

$$\texttt{vol}_t = \alpha^{(V)} \, |p_t - p_{t-1}| + (1 - \alpha^{(V)}) \, \texttt{vol}_{t-1} \tag{6.2}$$

with exponential decay $\alpha^{(V)}$ which we typically set to 0.02. The normalised price series will be mainly used when analysing and comparing different returns distributions.

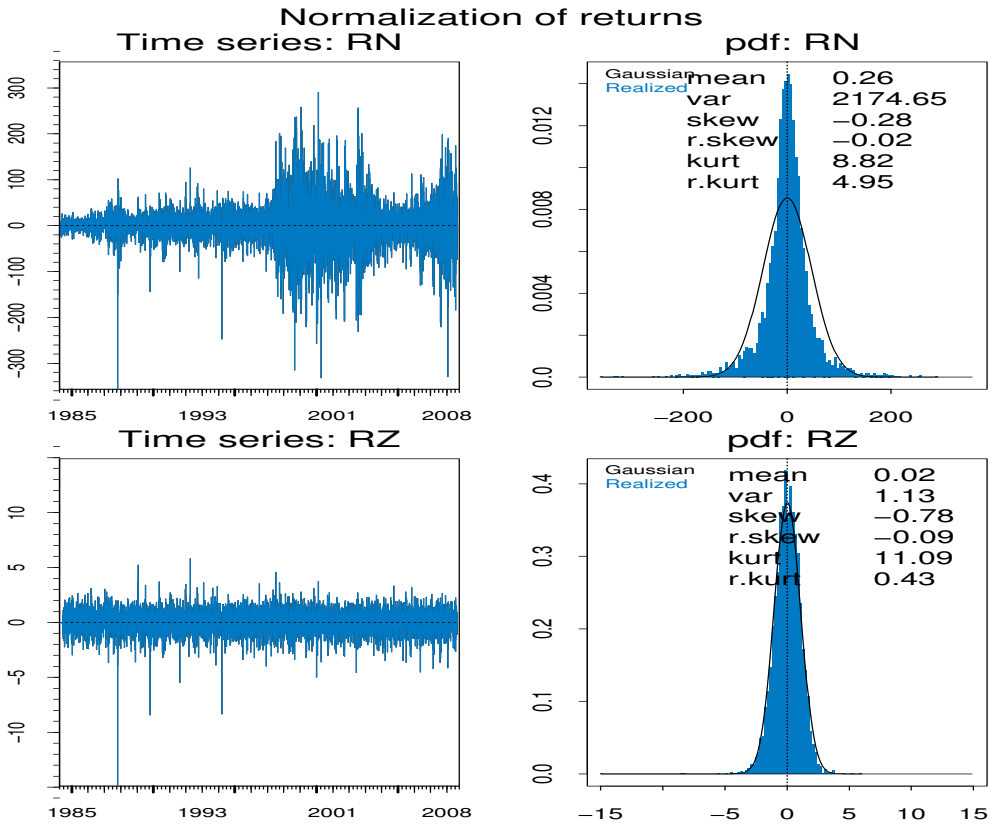

Figure 6.1: Normalisation by the rolling volatility. Top row, Simple price differences $\boldsymbol{y}$. Bottom row, price differences divided by rolling measure of volatility, $\boldsymbol{y}^{(v)}$.

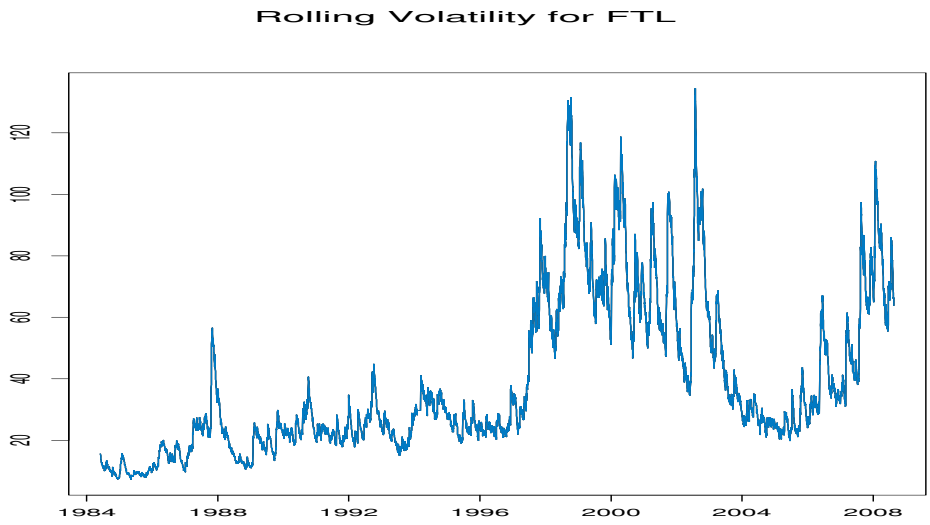

Figure 6.2: Rolling Volatility Measure of daily returns, FTSE future

6.3 Returns Distribution Statistics

We have established that we can investigate financial markets' dynamics by studying the price process and in particular the daily price returns which represent the evolution of the market through time. We now want to investigate which statistics we can compute to best describe a market returns distributions. The objective, as we discussed in chapter 1, is to measure the most relevant features of each empirical distribution so that we can characterize each market and have enough informative variables to guide the clustering assignment.

6.3.1 General Descriptive Statistics

The general descriptive statistics are used to describe the basic features of the returns' distribution, like the one in Figure 6.3. They provide simple summaries about the relevant aspects of its shape and together with simple graphics analysis they form the basis of virtually every quantitative analysis of data. We briefly describe all the statistics we compute grouped by the feature of the distribution they are trying to measure.

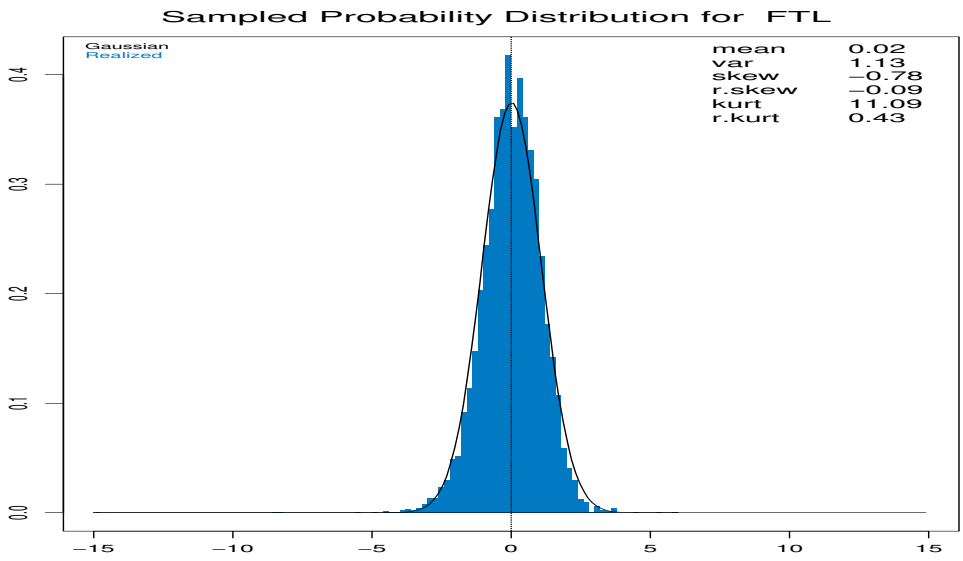

Figure 6.3: Sampled probability distribution for normalised FTSE futures returns.

Location. The distribution of markets returns is generally assumed to be centered around zero. In reality we can expect a marginally positive drift to accommodate for inflation and structural long term growth of the economy. The location of a distribution is indicated by the `Mean`, and `Median`. To exemplify how these simple statistics can characterize clusters of market, in Figure 6.4 we show the boxplot of sample mean and median grouped by the macro sectors. Even if our aim with the present study is to refine this partition, we note already that different sectors show distinct typical location.

Dispersion. It measures the dispersion of the population around its mean. In finance it is first of all considered a measure of risk where higher dispersion means more uncertainty and more chances of an adverse outcome. In the present thesis we consider the sample volatility measured as `Variance` and `Stdev`. In Figure 6.5, we can see how different sectors might be characterized by a different volatility level.

Range. The range statistics give information about the support of the distribution. Besides the `Minimum` and `Maximum` a more conservative and robust mea-

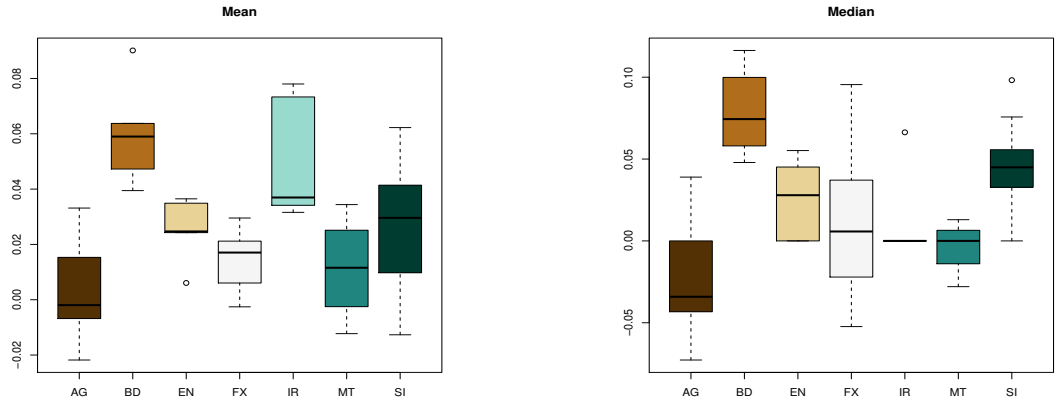

Figure 6.4: **Location**: Boxplot of sample `Mean` and `Median` of daily normalized returns, $\boldsymbol{y}^{(v)}$, for each market grouped by sector.

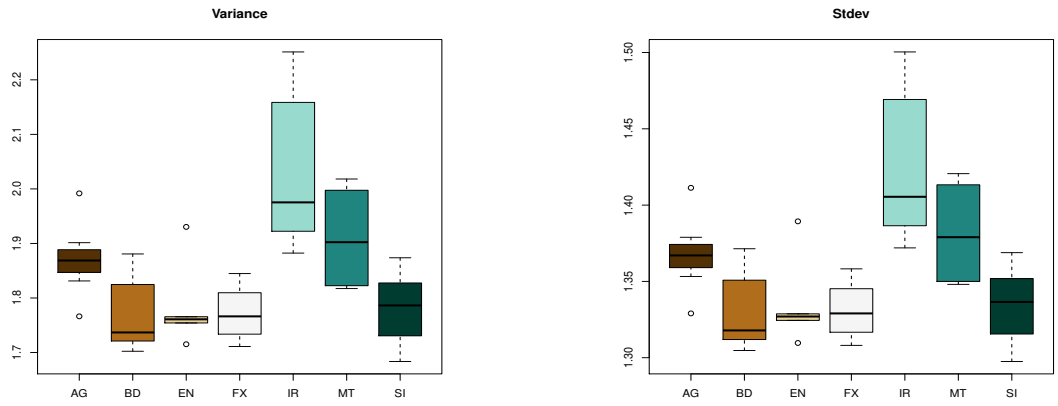

Figure 6.5: **Dispersion**: Sample Variance and Standard deviation of daily returns, $\boldsymbol{y}^{(v)}$, for each market grouped by sector.

sure of the range of a distribution is the interquartile range, `iqrt`, computed as the difference between the first quartile, `FirstQuartile` and the third quartile, `ThirdQuartile`. It is more robust because it avoids possible distortions introduced by outliers.

Similarly, the interquantile range, `InterQuantile`, is the difference between the 99.5th percentile and the 0.5th percentile. Whilst the interquartile range, `InterQuantile`, excludes any information contained in the tails of the distribution, the interquantile tries to filter out only the very extreme observations.

Asymmetry. The asymmetry of the distribution is measured by the skewness. The `Skew` is the third moment of the distribution about its mean. A more robust measure of skewness can be computed using quantiles to filter out possible spurious outliers and errors in the data: $\texttt{RobustSkewness} = (q_{0.5} + q_{99.5} - 2q_{50})/(q_{99.5} - q_{0.5})$.

Kurtosis. The kurtosis is the ratio of the fourth moment of distribution about its mean, μ_4, and the squared variance. For a Gaussian distribution its value is 3. It is normal practice to quote the excess kurtosis as a measure of the distribution's distance from normality. A positive value means that more of the variance is generated by infrequent extreme events as opposed to frequent modestly sized deviations. Such a distribution is called leptokurtic and, relative to a Gaussian distribution, shows more mass near the mean and fatter tails. If the opposite is true, it is called platokurtic and exhibits relative smaller peak and thinner tails. The `Robust Kurtosis` statistics offers some advantages. It is less effected by extreme realisations in the data which sometimes might not be genuine market moves but only error spikes. The robust kurtosis compares the estimated quantiles q of the sampled distribution versus the theoretical quantiles z of a normal distribution: $\texttt{RobustKurtosis} = (q_{99.5} - q_{0.05})/(q_{75} - q_{25}) - (z_{99.5} - z_{0.05})/(z_{75} - z_{25})$.

Tail Shape. Measuring the main features of the tail shape of a distribution can yield to very interesting insights in the properties of the markets' dynamics. As shown in Figure 6.6 we can use a Generalized Pareto Distribution (GPD) to model the tail of the daily returns. Using the standard GPD notation described in Appendix 6.A.1, the two relevant parameters we want to estimate from historical data are the scale parameter and the shape parameter ξ. The parameters can be

estimated via maximum likelihood estimation.

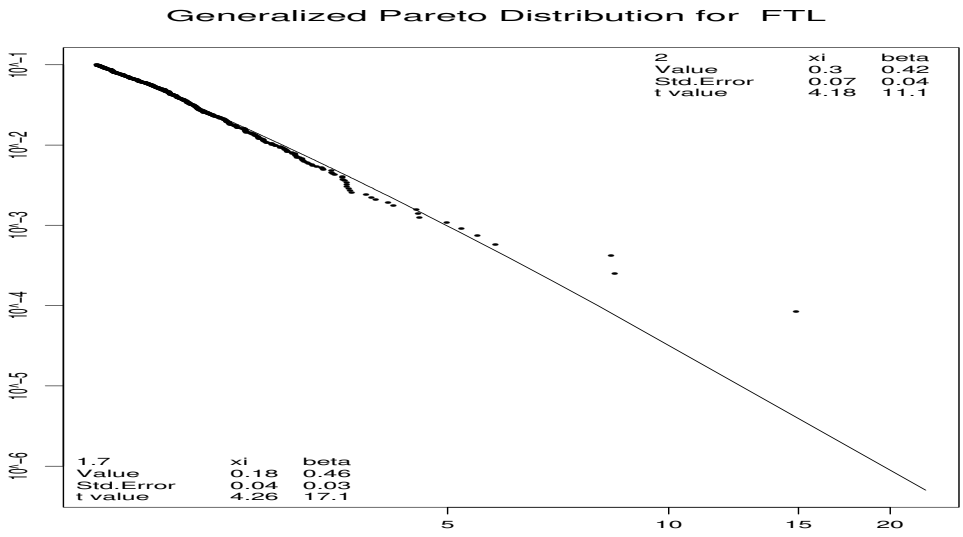

Figure 6.6: Fitted Generalised Pareto Distribution on joint lower and upper tail, where ξ has been estimated by maximum likelihood with threshold u set at the 95th percentile.

Alternatively, we can follow a non parametric approach, first proposed by Hill (1975), as shown in Figure 6.7. In this case the shape parameter ξ is approximated by averaging the tail slope over a reasonable interval as suggested by Gopikrishnan et al. (1999).

It is worth pointing out that we can obtain a more accurate characterisation of the markets' return distribution by studying the upper and lower tail separately. According to general belief markets fall faster than they rise since they are more likely to overreact to negative news than the other way around. We would then expect to find confirmation of this asymmetry by observing that the left tail of negative returns is heavier than the right tail of positive extreme events. Looking at Figure 6.8 we only find partial evidence that this is in fact the case.

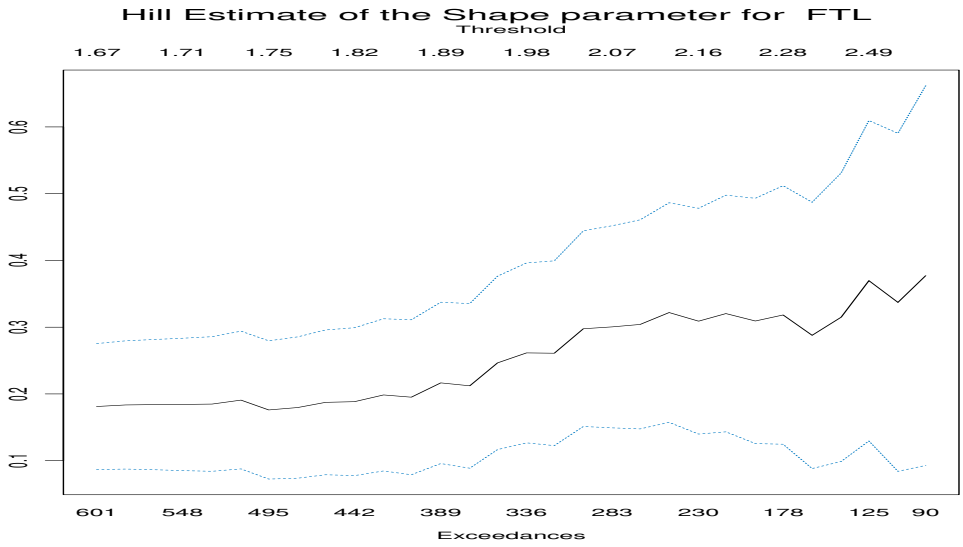

Figure 6.7: Hill estimator of the tail shape parameter ξ for increasing thresholds.

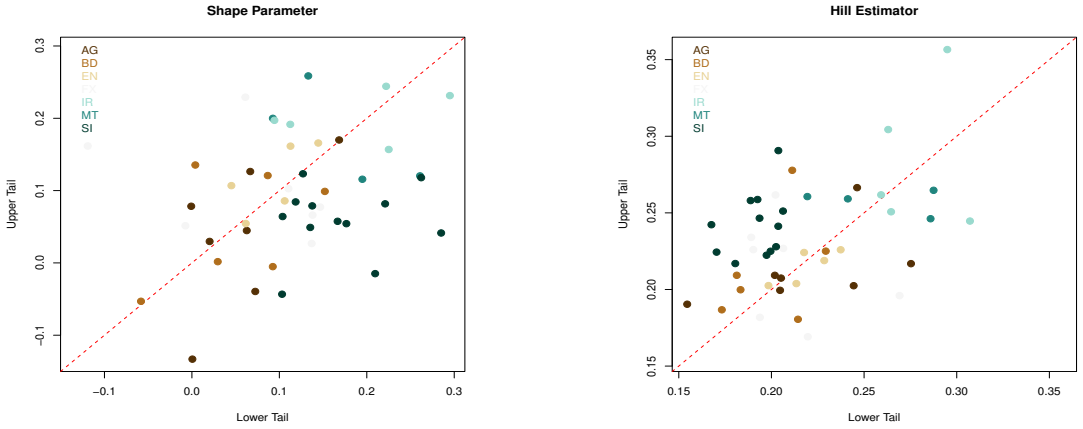

Figure 6.8: **Lower Vs Upper tail**: Tail shape indexes ξ estimated separately for positive and negative returns. Each point corresponds to a different markets and the colour coding represents the macro sector that market belongs to.

6.4 Returns Series

In the previous section we have reviewed some of the main statistics commonly used to describe market returns distributions. We now want to further characterize each markets dynamics by measuring the relevant properties of time series of returns where the temporal structure is preserved.

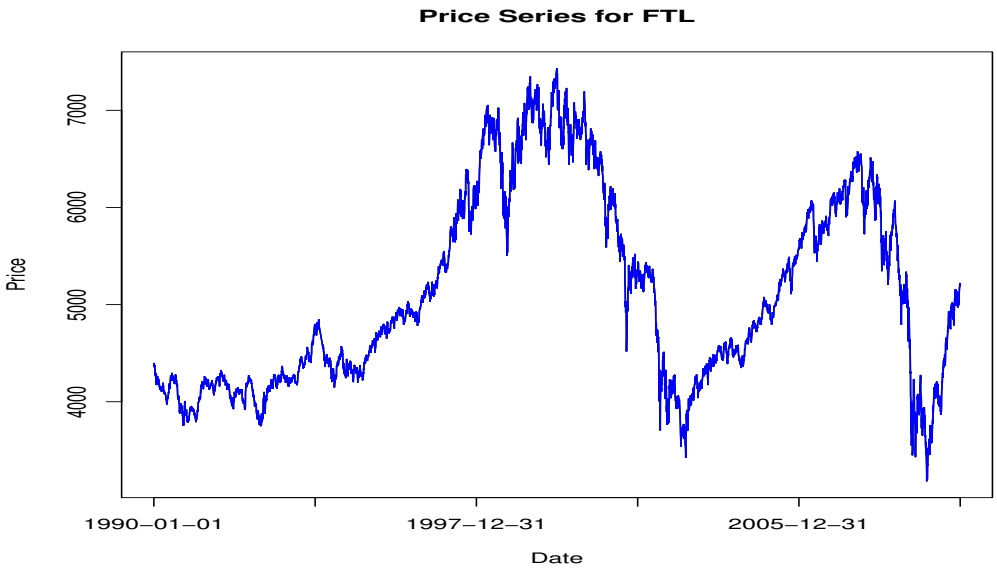

Figure 6.9: Price Series of rolled FTSE 100 Future.

It is worth pointing out that according to the efficient market hypothesis, prices should instantaneously discount, i.e. price in, all the public information as soon as it becomes available. Therefore, there should not be any trace of long memory in returns series since only the arrival of new information which is completely random could justify any price change. A simple general model for price processes in this case would assume that it can be represented as:

$$p_t = \mu_{t-1} + p_{t-1} + \epsilon_t, \quad t = 1, \ldots, T$$

where p_t is the price at time t, μ the mean change or drift and ϵ_t a random error and the return $y_t = \Delta p$ at time t is simply $y_t = \mu + \epsilon_t$.

In reality, the flow of relevant information does not appear to be completely

random and also the price moves seems to be not independent. More formally, it is reckoned that there are three degrees of market efficiency that correspond to three different types of random walks :

- RW1 $\epsilon_t \sim$ i.i.d. $(0, \sigma^2)$
- RW2 ϵ_t is an independent process (allows for heteroskedasticity)
- RW3 ϵ_t is an uncorrelated process (allows for dependence in higher order moments)

Several theoretical processes have been considered in the literature to conform to the different market efficiency hypotheses. Among these, we should mention fractional Brownian motion which has been proposed to model the price process when there is evidence of long-range dependence. In Appendix 6.A.2 we review the important properties of this and other theoretical processes that we will refer to in the following discussion.

Before describing the relevant statistics and how to compute them, let us recall that there are different ways of ordering the collection of returns $\{\ldots, y_{t-1}, y_t, y_{t+1}, \ldots\}$ depending on the index and frequency chosen. Typically, for a constant and smooth process, discretization is obtained by sampling at regular intervals. For financial price series we have three alternative domains that we can use to index and order the records: physical time domain, volume domain and transaction domain. In our following discussion we will use the time domain, but we consider the two other options as an interesting extension of the present work.

6.4.1 Return Series Statistics

We have detailed how we will extract information from the returns. Here we review, grouped by the particular feature they try to measure, the relevant statistics that we are going to compute from returns series in order to cluster markets.

Temporal Autocorrelation. As a preliminary test we will compute the basic autocorrelation of the returns series to verify whether at least the weak form of

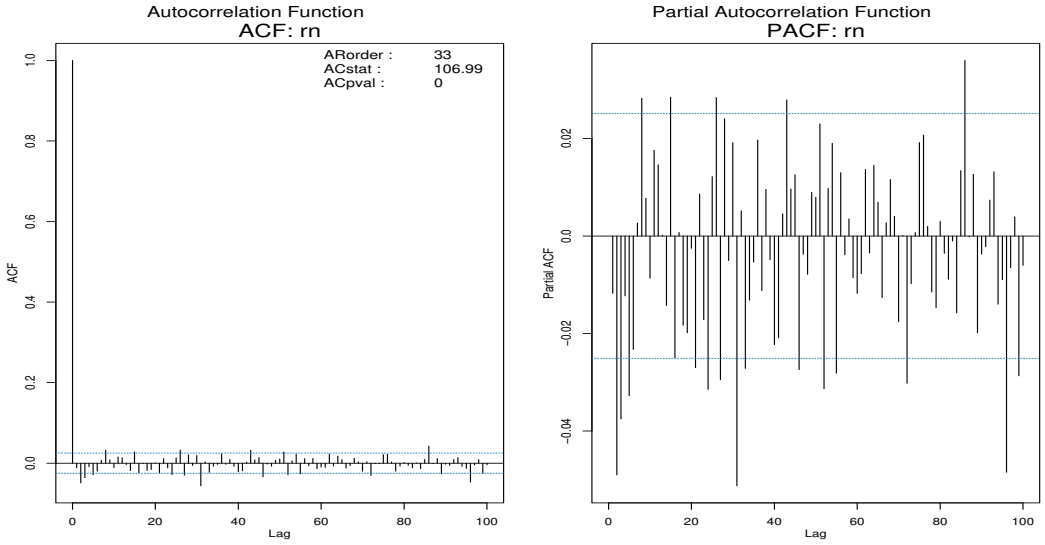

Figure 6.10: Autocorrelation and Partial Autocorrelation function.

market efficiency RW3 holds true. We can see an example of sample autocorrelation and partial autocorrelation function for a financial market in Figure 6.10.

Another statistic we compute to quantify the degree of linear dependence in return series is the $Q-$statistic which is designed to detect departures from zero autocorrelation in either direction and at all lag, see Campbell et al. (1997) for an example of an application to financial markets. In other words, we test the null hypothesis $H_0 : y_t \sim WN(0, \sigma^2)$ where WN denotes a white noise process. The $Q-$statistic proposed by Box and Pierce (1970) is obtained by summing the squared autocorrelations

$$\mathtt{Q}_m = T \sum_{j=1}^{m} \hat{\rho}_j^2$$

which is asymptotically distributed as χ^2_m. For finite samples an unbiased test statistic, called modified $Q-$statistic, was introduced by Ljung and Box (1978).

Scaling of Volatility. If a market satisfies the random walk hypothesis, then the variance of its returns should be a linear function of the time interval (Hamilton, 1994; Campbell et al., 1997). The purpose of the variance ratio test is precisely to

verify whether this hypothesis is confirmed

$$\texttt{vrt}(j) = \frac{\text{Var}(y_t^{(j)})}{j \cdot \text{Var}(y_t)}$$

where $y_t^{(j)} = y_{t-j+1} + \ldots + y_t$ and y_t is the return at time t. Note that a value $\texttt{vrt}(j) < 1$ denotes a mean reverting process, whereas $\texttt{vrt}(j) > 1$ indicates that the process is mean averting and the variance grows more than expected with the time interval. For more details on the modified variance ratio test we refer the interested reader to paper by Lo and MacKinlay (1988). In plot 6.11 we can see an example of `vrt` statistic computed for an increasing return interval j.

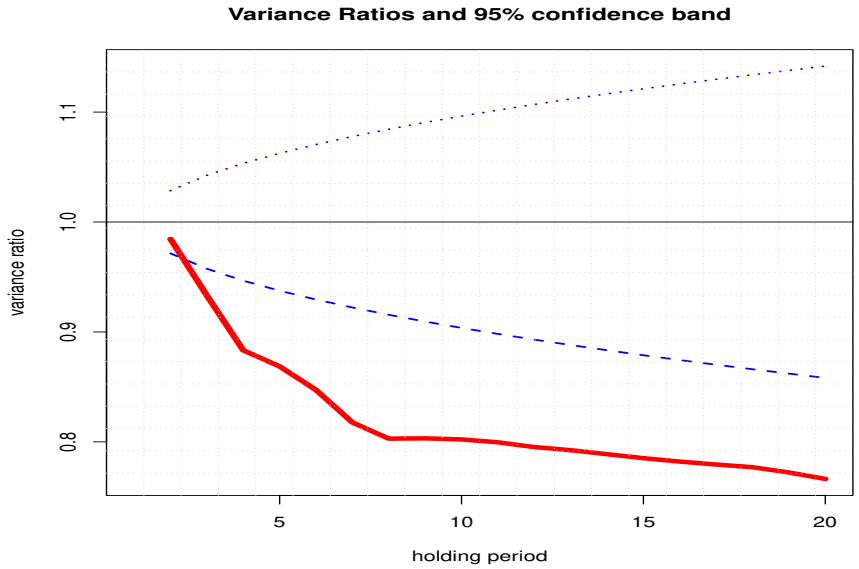

Figure 6.11: Variance Ratio Test for FTSE 100 Future. The price process appear to be mean reverting.

Long Memory. Another important aspect of markets returns, which is particularly interesting to the investment community, is the degree of persistence of the observed price changes. We have implemented different methods, either in the time domain or in the frequency domain, to quantify the level of persistence in returns series. In the time domain, long memory manifests itself as hyperbolic decaying autocorrelation functions. Whereas, in the frequency domain, the spectrum

appears to have high power at low frequencies.

The rescaled range statistic, `rst`, is computed to test the null hypothesis that the price increments are random. The original notion of R/S statistic was proposed by Hurst (1951) as a way of measuring the long term behaviour of the Nile river floods, but it has since found application also in finance (Peters, 1991; Sprott, 2003). The modified `rst` statistic that we quote in our analysis is the version suggested by Lo and MacKinlay (1988):

$$\texttt{rst} = \frac{1}{\hat{\sigma_T}(j)} \left[\max_{1 \le k \le T} \sum_{l=1}^{k} (y_l - \bar{y}) - \max_{1 \le k \le T} \sum_{l=1}^{k} (y_l - \bar{y}) \right]$$

the difference with Hurst's original version lays entirely on the estimate of the volatility $\hat{\sigma}_T(j)$ as computed by Newey and West (1987). The advantage of $\hat{\sigma}_T(j)$ is that it takes into account not only the sums of squared deviations of the individual terms y_t, but also the autocovariances up to lag j. The test statistics `rst` should scale proportionally to the power of the time interval T, $\texttt{rst} \approx (aT)^H$ where a is just a normalizing constant and H is the Hurst Exponent. Note that, in the absence of a long-run statistical dependence, H is $1/2$ therefore when returns are generated by an independent process with finite variances we obtain $\texttt{rst} = (\pi T/2)^{1/2}$. Gneiting and Schlather (2001) found that the following relation $H = \frac{\log(\texttt{rst})}{\log(T)} = d + \frac{1}{2}$, where d is the fractal dimension, suggests a practical way to estimate the Hurst exponent H. As shown in Figure 6.12, they propose to regress its value by plotting the $\log \texttt{rst}$ versus the $\log T$.

The `GPH` test was proposed by Geweke and Porter-Hudak (1983) as an alternative approach to test for the presence of long memory is based on Wold's spectral representation (Hamilton, 1994). Similarly the periodogram method, `prd`, and the Whittle's method, `whd`, provide a practical method to estimate the fractal dimension d by decomposing the price process in its frequency domain.

The **Generalized Hurst Exponent** (GHE) described in Barabási and Vicsek (1991); Di Matteo (2007); Di Matteo et al. (2004), allows a richer analysis of persistent dynamics. By computing the q-order moments of the distribution of the

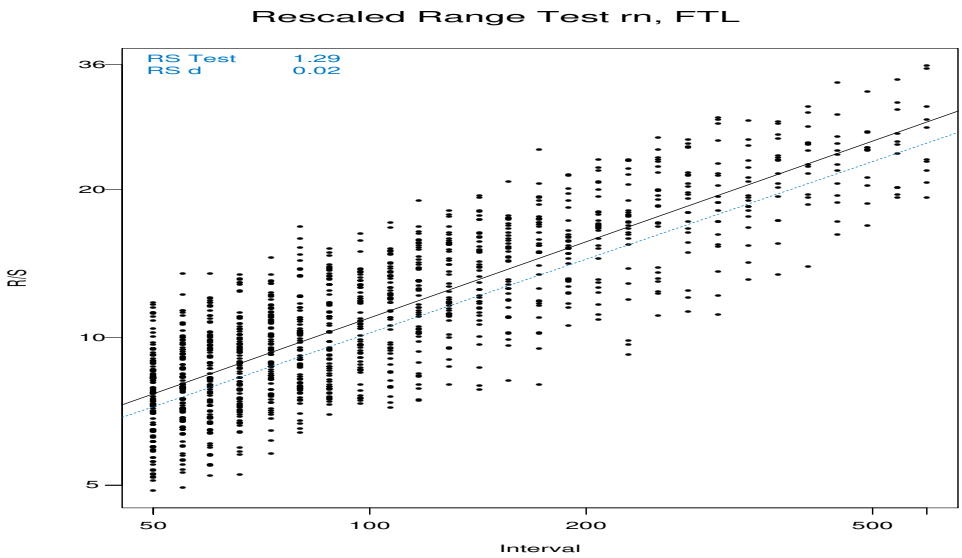

Figure 6.12: Rescaled Range Test

increments, $K_q(\delta) = \frac{\mathbb{E}(|Y(t+\delta)-Y(t)|^q)}{\mathbb{E}(|Y(t)|^q)}$ where δ is the time-interval, we are much less sensitive to the outliers than if we were using maxima/minima. The generalized Hurst exponent $H(q)$ can be defined as a property of the scaling behaviour of $K_q(\delta)$

$$K_q(\delta) \sim \left(\frac{\delta}{\nu}\right)^{qH(q)}$$

In Figure 6.13 we see an example of how we proceed to empirically estimate the GHE from the q-order moments.

6.5 Data Matrix

In previous sections of this chapter we have reviewed some of the relevant features of the distribution and time series of returns that we believe can help us to characterize the underlying price dynamics of each market.

Note that at this point we have not tried yet to isolate the most informative features, but we consider all as potentially relevant for clustering. For each of the $n = 46$ financial markets we have data for, we compute $p = 81$ statistics which we

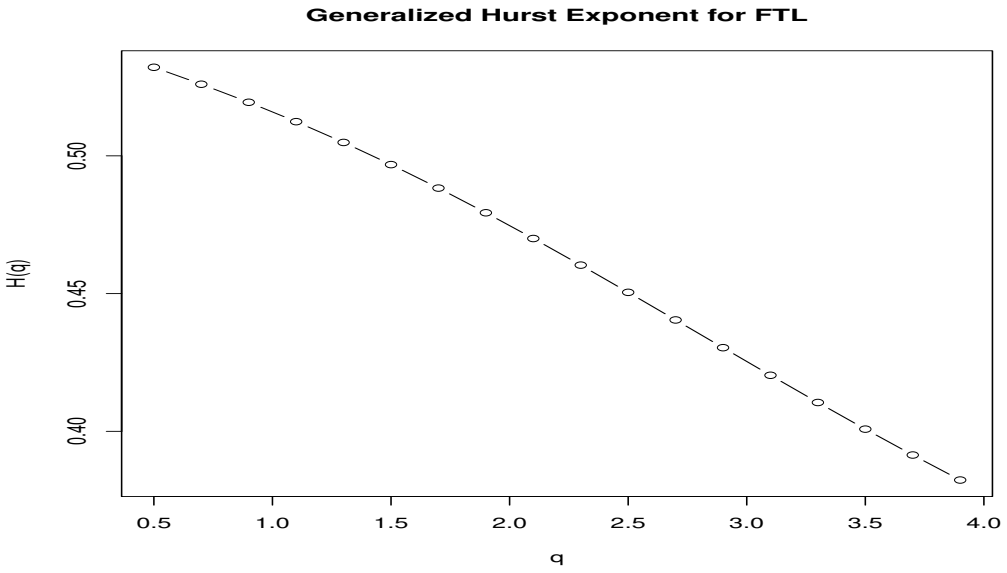

Figure 6.13: Generalized Hurst Exponent estimate for FTL.

arrange in a 46×81 data matrix. We expect that both the models we introduced in chapter 3 and 4 will be able to automatically select the important variables and produce a reasonable cluster assignment of markets.

As a preliminary overview, in Figure 6.14, we plot the correlation matrix of the different statistics computed over all markets. We can see that some variables are, as expected, highly correlated, for example kurtosis and robust kurtosis, or variance and interquantile range.

Response Variable

Let us recall that the ultimate goal of the study is to find a more appropriate systematic trading strategy whose parameters can be robustly calibrated on clusters of similar markets.

To verify that the markets' return features we have described are related to the trend following strategies we are interested in, we compute a risk adjusted measure of investment performance as response variable. We adopt a simple moving average crossover to generate buy or sell signals and target a constant risk profile by scaling

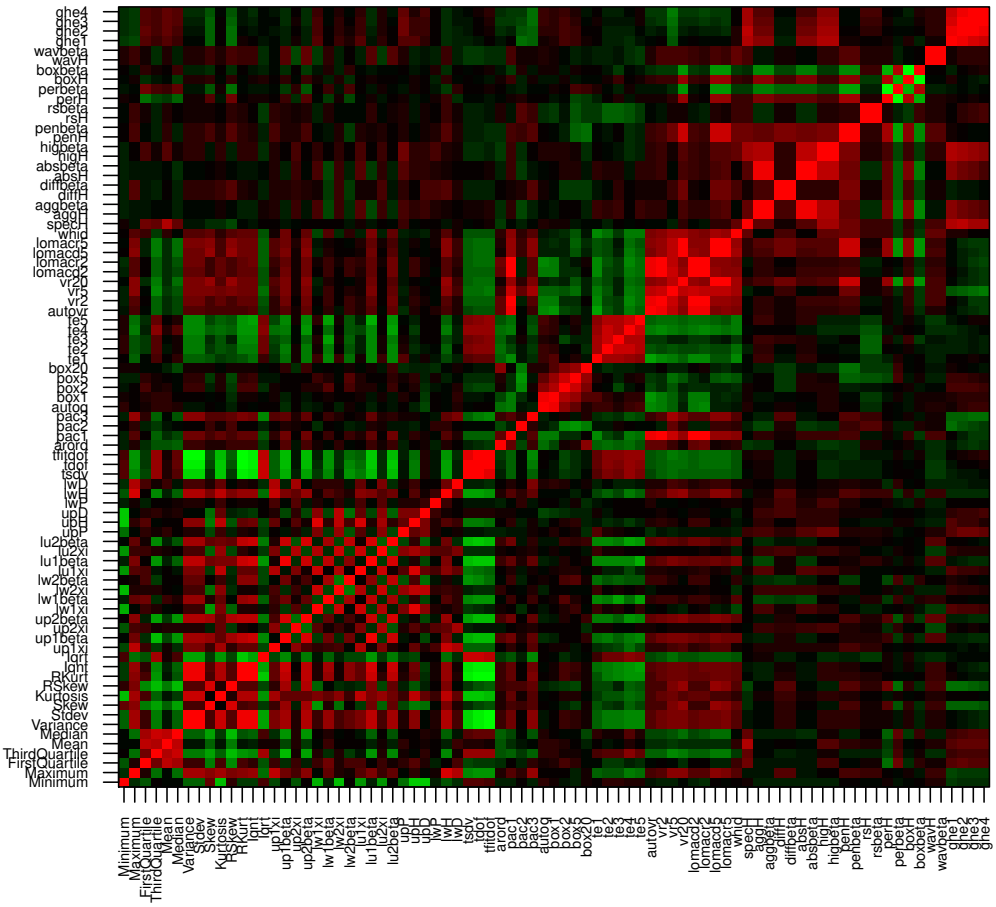

Figure 6.14: Correlation between different statistics computed on all markets.

positions according to a rolling volatility measure. Given a time series of prices $\{p_t\}_{t=1}^{T}$, and Ania's value $\texttt{EMA}_1 = p_1$, the exponential moving average at time $t > 1$ is

$$\texttt{EMA}_t = \alpha\, p_t + (1-\alpha)\texttt{EMA}_{t-1}$$

where α represents the degree of exponential decay of the weights associated to older prices. The value of α determines the speed at which the exponential moving average reacts to a new recorded price and ultimately how close it tracks the price process. To generate our position signal we compute a fast $\texttt{EMA}^{(\texttt{F})}$ and a slow

$\texttt{EMA}^{(\texttt{S})}$ by fixing $\alpha^{(F)} = \{0.03\}$ and $\alpha^{(S)} = \{0.01\}$ respectively. A buy signal is then generated every time the fast moving average crosses from below the slow moving average; conversely, a sell signal is given when it crosses from above. The position, pos, is then held proportional to the difference between the two moving averages,

$$\texttt{pos}_t = (\texttt{EMA}_\texttt{t}^{(\texttt{F})} - \texttt{EMA}_\texttt{t}^{(\texttt{S})})/\texttt{vol}_\texttt{t}$$

where vol is a measure of the rolling volatility computed as from (6.2) with fixed $\alpha^{(V)} = 0.02$. The point of scaling the position proportionally to the volatility of the price process, is to automatically adjust the risk of our exposure to the perceived uncertainty of the market. In practice, when the volatility increases we would scale down our positions. In Figure 6.15 we report, as an example, the diagnostic plots of a systematic trading strategy applied to the FTSE 100 future. The annualised Sharpe Ratio (Sharpe, 1966), in the bottom plot, measures the average return per unit of risk and it is computed from the sequence of the daily profits and losses $r_t = \texttt{pos}_{1-t} \times (p_t - p_{t-1})$

$$\texttt{sr} = \frac{250/T \sum_{t=1}^{T} r_t}{\sqrt{250 \operatorname{Var}(\boldsymbol{r})}}$$

where 250 is the number of working days in a year.

We have remarked that our goal is to verify whether the commonly accepted partition of financial markets in macro sectors is justified. In Figure 6.16 we find some evidence that the same strategy applied to all markets generally returns a better or worse Sharpe ratio depending on the macro sector the market is in.

6.6 Penalised t Mixture Model

The first model we fit using the 46×81 data matrix described in Section 6.5 is the penalised mixture of Student's t components model. Since this is an unsupervised clustering method, at this point, we do not include in the fitting process the response variable, i.e. the Sharpe ratios. Our goal here is to find a reasonable objective clustering of the markets and simultaneously identify the informative

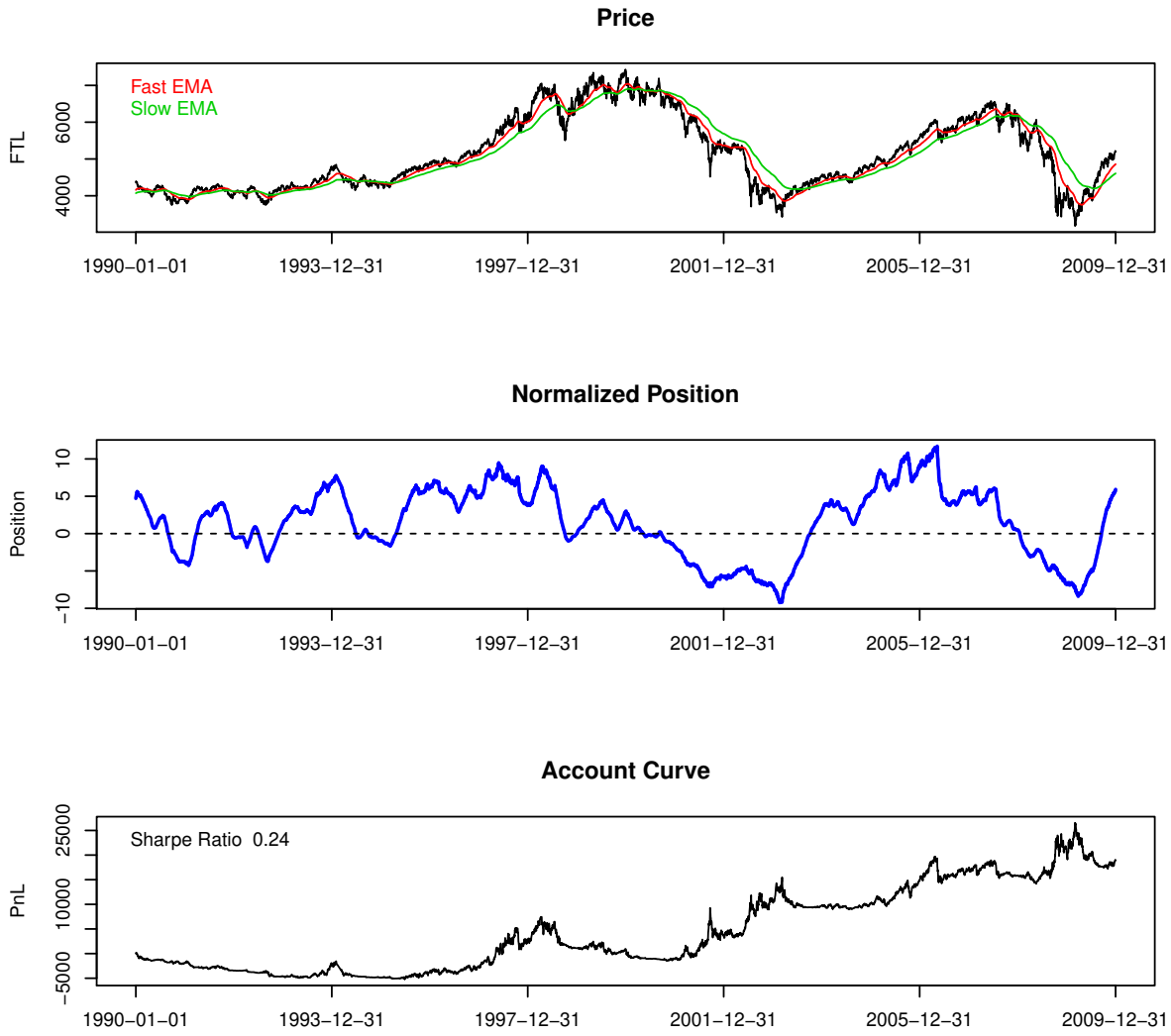

Figure 6.15: Systematic Trend Following Trading Strategy applied to FTSE 100 Future.

statistics that depict the relevant features of the price dynamics.

We know from section 3.4, that before we can proceed to estimate the model parameters, we need to set the number of clusters, K, and fix an optimal level of penalisation λ_μ and λ_σ. We follow the exact procedure illustrated in section 3.5.2 and fit a separate model for $K = 2, \ldots, 5$ where for each K we test all possible

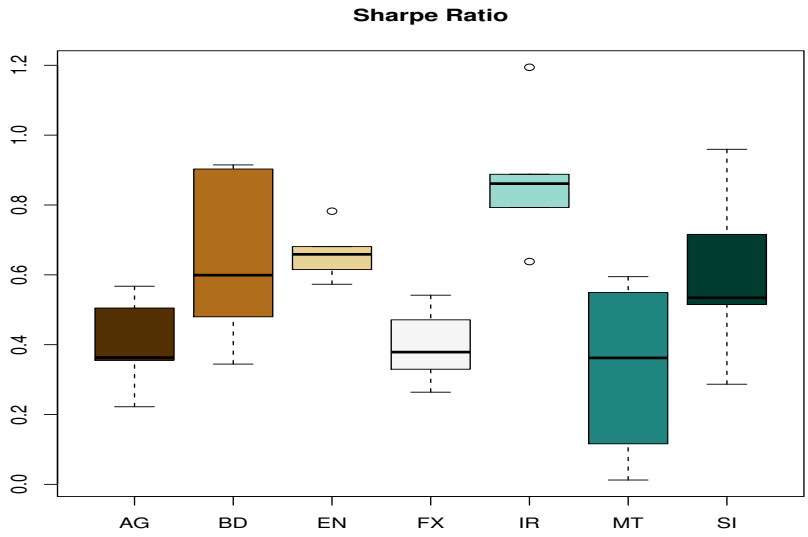

Figure 6.16: Sharpe Ratios of a simple trend following trading strategy applied to different markets grouped by fundamental macro sectors.

combinations of λ_μ and λ_σ in the interval $[0, \ldots, 10]$. In Tables 6.2, 6.3, 6.4 and 6.5 we illustrate the outcome of these procedure. The matrices report the distance in `BIC` units from the lowest `BIC` level reached by all combinations. We note that despite some discontinuities there is a fairly clear indication of the region where the optimal model is.

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	372	349	364	421	458	475	510	569	689	697	696
1	144	38	64	142	175	205	232	295	364	387	398
2	NA	12	39	92	146	176	209	231	264	298	403
3	NA	0	57	87	145	165	201	216	250	283	292
4	323	39	72	115	161	177	201	229	248	273	292
5	NA	123	97	121	156	179	204	228	253	278	292
6	NA	253	128	122	166	192	212	237	259	276	293
7	NA	175	152	160	176	196	218	242	267	286	304
8	2264	231	185	171	183	211	229	251	269	293	292
9	2325	70	224	195	200	218	233	252	273	289	315
10	2404	577	268	219	225	241	238	256	279	294	312

Table 6.2: Model Selection, grid search assuming $K = 2$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

In Figure 6.17 we can see the smoothed contour levels of the `BIC` index on the grid of λ_μ and λ_σ for every K. Note that, in line with the conclusions drawn from

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	798	706	746	783	834	863	685	929	969	1002	1023
1	550	230	409	328	532	412	452	415	530	709	727
2	798	139	376	263	475	212	556	400	646	429	469
3	874	216	360	213	447	298	335	340	352	606	399
4	NA	0	203	184	283	381	491	554	578	346	363
5	2681	465	286	249	335	404	304	346	387	408	610
6	1027	136	433	321	272	318	519	349	609	608	622
7	1185	94	427	352	432	447	450	372	407	413	428
8	3047	267	418	370	378	437	359	372	418	611	498
9	1200	807	492	434	413	409	376	424	475	477	624
10	2254	803	630	421	396	429	447	463	475	624	499

Table 6.3: Model Selection, grid search assuming $K = 3$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	1269	1209	1334	1400	884	1525	941	974	1564	1030	1669
1	713	681	730	788	802	595	775	701	706	1104	1096
2	1420	552	563	611	693	737	597	861	868	794	962
3	2909	693	540	630	658	382	789	661	803	657	933
4	1739	913	604	531	514	634	354	618	861	883	774
5	1901	1127	659	432	467	612	682	670	697	713	929
6	3110	1220	761	475	476	638	723	698	871	892	789
7	1756	569	673	700	468	691	720	728	808	775	846
8	1856	1037	756	692	652	607	629	659	807	820	972
9	1474	0	565	743	659	683	688	713	764	829	838
10	2233	1408	809	715	681	700	759	687	808	827	851

Table 6.4: Model Selection, grid search assuming $K = 4$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

$\lambda_\mu \backslash \lambda_\sigma$	0	1	2	3	4	5	6	7	8	9	10
0	945	1312	1644	1547	1626	1673	1911	1295	1181	1345	1258
1	1198	856	855	954	999	502	518	605	644	1153	725
2	1211	590	519	493	582	442	600	462	858	664	662
3	1304	241	455	520	548	407	350	377	740	637	613
4	1971	794	439	509	529	514	631	525	462	656	842
5	1612	1797	553	416	318	480	268	497	536	389	427
6	1956	1751	371	284	638	537	514	556	359	373	664
7	NA	25	483	596	226	438	480	529	666	688	707
8	3111	1621	890	481	458	531	652	565	691	690	506
9	2183	253	727	515	394	464	500	624	682	696	718
10	2250	0	720	650	538	435	570	245	608	707	718

Table 6.5: Model Selection, grid search assuming $K = 5$. Difference in BIC units from best penalisation level λ_μ and λ_σ.

experimental scenarios, the optimal level of penalisation is fairly robust irrespective of the number of components and that the joint penalisation is always a better option than penalising only the location or the dispersion parameter.

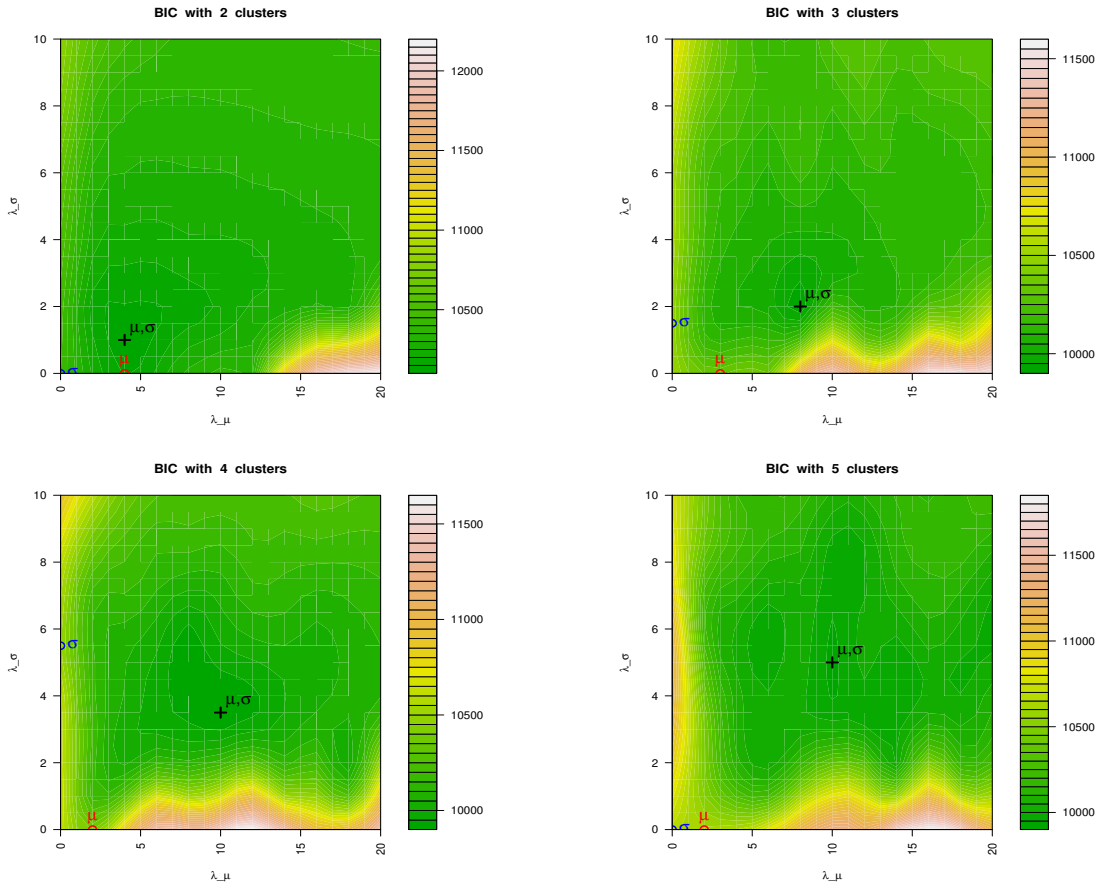

Figure 6.17: **Optimal Penalisation**: Using BIC criteria to identify the optimal level of penalisation λ_μ and λ_σ for $K = 2, \ldots, 5$.

Once we have established the optimal λ_μ^* and λ_σ^* for each $K = 2, \ldots, 7$, we implement the subsampling routine described in Section 3.4.1 and retain only the variables whose selection probability is above the threshold $\tilde{\pi} \geq 0.7$. We then compare the likelihood of each fitted model and follow the `BIC` criterion to choose the best one. In Table 6.6 we report the results of the model selection procedure and observe that the `AIC` and `BIC` criteria agree in indicating $K = 4$ as the best model, whereas the likelihood criterion alone would have chosen a less parsimonious model.

To asses how robust our clustering assignment is and how different the results would be if we had chosen a different value K, in Figure 6.18 we plot the heatmap

# Clusters	LLIK	AIC	BIC
2	-1654	3563	3795
3	-1417	3217	3566
4	-1271	3053	3520
5	-1212	3062	3646

Table 6.6: Model Selection PTM

of the ARI index computed for each pair of tested models from Table 6.6. We note that the two model $K = 4$ and $K = 5$ arrive to very similar conclusions as their ARI is 68%. This result confirms why their BIC score being so close.

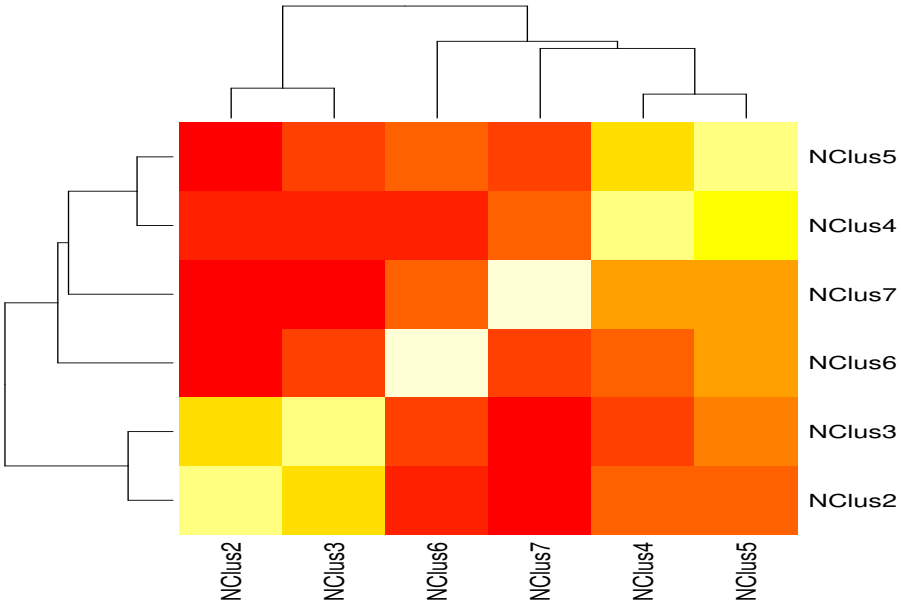

Figure 6.18: Cluster agreement of each pair of models for $K = 2, \ldots, 7$ measured by Adjusted Rand Index.

6.6.1 Clustering Results

Based on the indications of the model selection procedure, we fit a mixture of four components, $K = 4$, with a penalisation of the location parameters equal to $\lambda_\mu = 9$ and $\lambda_\sigma = 5$ for the dispersion parameters. In Table 6.7 we report the results of the clustering assignment. We can see that there is some degree of overlapping with

the fundamental sector partition. The adjusted rand index is $\texttt{ARI} = 28\%$ which is fairly high considering that the number of macro sectors is 7 while we only assume 4 clusters.

Looking at the results in more detail, the first thing to notice is that all interest rate markets are kept as a separate cluster with the only exception of the Japanese bond future which is known to be an odd market compared to other bonds. It is interesting, on the other hand, to see that the most liquid and efficient markets, such as the majority of the stocks, currencies and bonds, are considered as a single homogeneous group. We can also say that while the energies and agricultural markets are allocated to cluster A and B respectively, the metals have been split with some insight between the industrial metals (aluminium and copper), and precious metals (gold and silver).

Cluster A		Cluster B		Cluster C		Cluster D	
MARKET	SECTOR	MARKET	SECTOR	MARKET	SECTOR	MARKET	SECTOR
Aluminium	METALS	Gold	METALS	Jap.Bond	BONDS	Wheat	AGS
Copper	METALS	Silver	METALS	Eurodollar	IRATES	Cotton	AGS
Corn	AGS	Soyabeans	AGS	S.Sterling	IRATES	EURUSD	CCY
Crude.Oil	ENERGY	Sugar	AGS	Euribor	IRATES	AUDUSD	CCY
Heat.Oil	ENERGY	Coffee	AGS	Euroyen	IRATES	CHFUSD	CCY
Gasoline	ENERGY	Cocoa	AGS	Aus.T.Bills	IRATES	GBPUSD	CCY
Gas.Oil	ENERGY	JPYUSD	CCY			EURJPY	CCY
		Taiwan	STOCKS			CADUSD	CCY
		Nat.Gas	ENERGY			SP500.Fut	STOCKS
						SP.Canada	STOCKS
						FTSE	STOCKS
						DAX	STOCKS
						CAC40	STOCKS
						NIKKEI	STOCKS
						TOKYO	STOCKS
						Aus.Spi	STOCKS
						EU.STOXX	STOCKS
						Hang.Seng	STOCKS
						Kospi	STOCKS
						10y.T.Notes	BONDS
						Gilts	BONDS
						Bund	BONDS
						Aus.10y	BONDS
						Cad.Bond	BONDS

Table 6.7: Cluster Assignment for $K = 4$

Variable Selection Results

In terms of variable selection, it is interesting to observe in Table 6.8 the ranking of the variables retained by the model after the resampling procedure, that is the variables whose selection probability is above the threshold $\tilde{\pi} \geq 0.7$. To asses how robust the results are we also report the ranking for the $K = 5$ model.

We note that the features that appear to be more informative for clustering is the distribution shape, measured by the kurtosis and standard deviation, together with the scaling property of the volatility, measured by the variance ratio test, `vrt`. One plausible explanation would be that kurtosis, and the scaling of volatility to some extent, are a proxy of the liquidity and efficiency of the market. It is generally accepted that the more evolute and efficient a market is, the faster any imbalance between demand and offer will be corrected, therefore reducing the possibility of extreme moves which are at the base of leptokurtic distributions. Similarly, the variance ratio test statistics could be used to distinguish between mean reverting and mean averting processes, where a mean reverting price process is again usually expression of an efficient and liquid market, like most of those in cluster D. In conclusion our proposed clustering seems to be mapping to some degree the efficiency of the different markets.

As a final independent check of the relevance of the partition we have identified, we plot in Figure 6.19 the boxplot of the Sharpe ratios of the markets in each of the proposed clusters. Even if Sharpe ratio has never been part of the fitting process we note a noticeable difference of its distribution across different clusters. This evidence seems to confirm the validity of the results we obtained and would suggest to further investigate the possible implications on real life trading.

These results, in fact, are to be expected since it is known that trend following strategies strives when there is higher volatility and extremes market moves. Being purely directional are best positioned to take advantage of unusually bigger markets returns, the same returns that increase the kurtosis of the distribution.

	4 Clusters		5 Clusters	
Rank	Variable	Selection π	Variable	Selection π
1	Kurtosis	0.948	vrt	1.000
2	RobustKurt	0.911	Kurtosis	1.000
3	StdDev	0.903	tDof	0.955
4	vrt	0.851	RobustKurt	0.955
5	tDof	0.851	box2	0.933
6	Maximum	0.851	tsdv	0.933
7	Iqnt	0.844	Iqnt	0.933
8	wavbeta	0.822	box1	0.888
9	wavH	0.822	Iqrt	0.888
10	box2	0.822	box5	0.866
11	box1	0.800	lwD	0.866
12	lwH	0.770	lwH	0.866
13	Iqrt	0.770	tfitdof	0.844

Table 6.8: Variable Selection. Highest ranking variables according to resampling method for model $K = 4$ with $\tilde{\pi} \geq 0.7$ and for model $K = 5$.

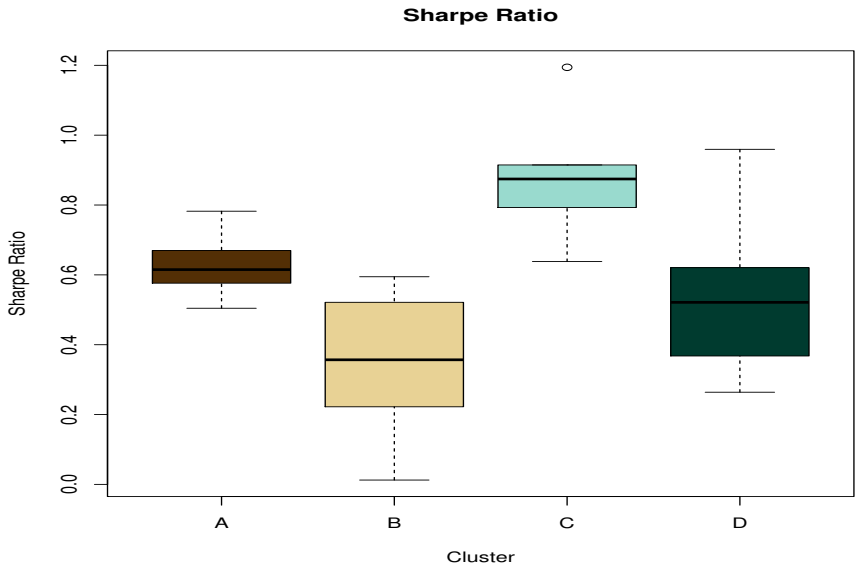

Figure 6.19: Sharpe Ratio of the same trend following trading strategy applied to markets clustered according to penalised t mixture model for $K = 4$.

6.7 Mixture of Lasso Regressions

We tested the penalised mixture of t distributions and proposed a new clustering of the financial markets. We now fit the Bayesian mixture of Lasso regressions and

compare the results of the two models.

In the present implementation of the mixture of regressions we will assume the same number of components $K = 4$ as we have found from the resampling procedure. Similarly, to have a more significant result, we reduce the column dimensions of the design matrix down to only those that have a selection probability above the threshold $\tilde{\pi} \geq 0.7$ as reported in the table 6.8. The observed dependent variable $\boldsymbol{y}$ is the Sharpe ratio of the simple moving average crossover strategy we described in section 6.5.

To fit the model we follow the sampling procedure described in Section 4.3.1 and execute ten thousand iterations of the PMCMC algorithm with adaptive resampling. As a preliminary check that the process has run as expected we record an acceptance rate of the proposed updates for τ, s of about 0.248, 0.262 respectively, which are in the correct range.

We should remark that to avoid the label switching problem common to every Bayesian mixture sampling procedure, we permute all possible labelling combinations of the components and choose the one that maximizes the adjusted rand index computed with respect to cluster assignment proposed by the penalised t mixture model. It is interesting to see in Figure 6.20 that there is in fact a reasonable agreement between the two methods with an average $\texttt{ARI} = 32\%$.

Whilst the explicit estimation of the regression coefficients was not part of the scope of this study, in Table 6.9 we quote the relative selection frequency of each variable across all PMCMC iterations. In practice we compute the average of the $\boldsymbol{\gamma}$ indicator vector over the ten thousands iteration we run. Since all of these statistics were previously selected by the resampling method applied to the `PTM`, we do not expect in this case to see a clear separation between some more informative and other less informative variables. The frequency of inclusion is fairly similar for all statistics. Note anyway that the variance ratio test statistics and the kurtosis are still positioned high in the table.

In Table 6.10 we report the empirical posterior probability of the label indicator variable z_i for $i = \{1, \ldots, n\}$ which indicates what it is the most likely cluster assignment for each market. It appears that in most cases there is a fairly clear

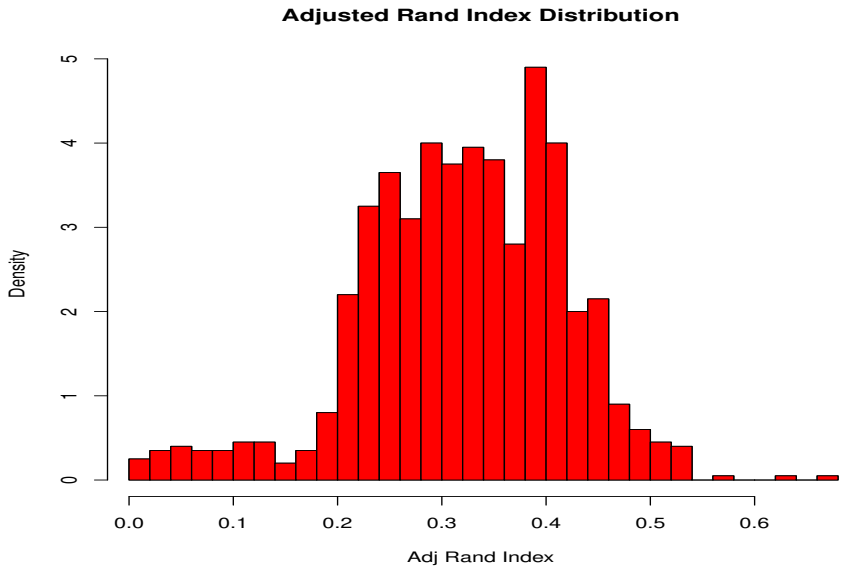

Figure 6.20: Adjusted Rand Index Distribution between sampled particles after every PMCMC iteration and proposed clustering by the penalised t mixture `PTM`.

indication of what cluster each market should belong to, see for example Aluminium. In some other there seems to be more uncertainty between two possible assignments, for example Soyabeans. To better asses the relevance of these conclusions, we compare the posterior probability versus the hard assignment of the `PTM` model. We note that generally there is only a partial overlapping, but that there is almost perfect matching with cluster B. A direct estimation of the regression coefficient could shade some light on what is the exact relation between each feature and the Sharpe Ratio of that market. To draw stronger conclusions for the other markets we would need further investigation.

6.8 Discussion

In this chapter we have investigated the real life problem of clustering financial markets based on some objective and measurable features of their price dynamics. We first computed a number of statistics that characterize the distribution and time series of daily returns and then fitted the two proposed mixture models we

Rank	Variable	Selection Frequency
1	Maximum	0.6475
2	wavbeta	0.5825
3	Kurtosis	0.5725
4	box1	0.5625
5	vrt	0.56
6	tsdv	0.535
7	Robustkurt	0.5325
8	wavH	0.5275
9	Iqrt	0.52
10	tdof	0.5125
11	Iqnt	0.51
12	box2	0.495
13	lwH	0.4825

Table 6.9: Variable Selection. Frequency each variable is selected.

presented in previous chapters.

The penalised mixture of t distributions provides some clear indications on how to cluster financial markets. This result seem reasonable and justifiable given the variables that have been selected as informative. As an independent check, we plotted the distribution of the Sharpe ratios by clusters and found significant evidence suggesting that there might be a link between the clusters we have identified and the profitability of a simple trend following strategy we have implemented. Further research could investigate the direct relation between the relevant statistics and the market behaviour triggering the trading signal and therefore suggest a better strategy to exploit it.

Using the smaller subset of statistics as explanatory variables and the Sharpe ratio as response variable, we then fit the mixture of Lasso regressions model. In this case the clustering results are more difficult to interpret but still confirm, to some extent, the indications from the `PTM` model.

In conclusion, we have been able to implement the proposed models to investigate a real life problem and obtain some promising insights both from the clustering and the variable selection point of view.

MARKET	SECTOR	PTM	Clus A	Clus B	Clus C	Clus D
Aluminium	METALS	A	78	2	19	1
Copper	METALS	A	28	12	13	47
Corn	AGS	A	32	15	20	33
Crude.Oil	ENERGY	A	56	10	24	10
Heat.Oil	ENERGY	A	57	8	30	5
Gasoline	ENERGY	A	60	7	26	7
Gas.Oil	ENERGY	A	54	5	22	19
Gold	METALS	B	3	50	9	38
Silver	METALS	B	10	37	49	4
Soyabeans	AGS	B	8	37	42	13
Sugar	AGS	B	5	54	4	37
Coffee	AGS	B	10	37	18	35
Cocoa	AGS	B	16	23	41	20
JPYUSD	CCY	B	10	29	23	38
Taiwan	STOCKS	B	14	32	25	29
Nat.Gas	ENERGY	B	8	32	36	24
Jap.Bond	BONDS	C	32	12	33	23
Eurodollar	IRATES	C	60	9	22	9
S.Sterling	IRATES	C	60	8	24	8
Euribor	IRATES	C	59	7	29	5
Euroyen	IRATES	C	34	21	27	18
Aus.T.Bills	IRATES	C	54	6	24	16
Wheat	AGS	D	50	6	28	16
Cotton	AGS	D	63	7	23	7
EURUSD	CCY	D	59	7	22	12
AUDUSD	CCY	D	63	3	26	8
CHFUSD	CCY	D	44	5	33	18
GBPUSD	CCY	D	63	6	24	7
EURJPY	CCY	D	58	7	24	11
CADUSD	CCY	D	61	6	24	9
SP500.Fut	STOCKS	D	58	6	26	10
SP.Canada	STOCKS	D	59	7	24	10
FTSE	STOCKS	D	62	6	22	10
DAX	STOCKS	D	61	6	23	10
CAC40	STOCKS	D	60	6	24	10
NIKKEI	STOCKS	D	57	6	26	11
TOKYO	STOCKS	D	57	12	24	7
Aus.Spi	STOCKS	D	63	7	23	7
EU.STOXX	STOCKS	D	61	10	27	2
Hang.Seng	STOCKS	D	65	10	21	4
Kospi	STOCKS	D	58	7	28	7
10y.T.Notes	BONDS	D	56	9	28	7
Gilts	BONDS	D	53	9	28	10
Bund	BONDS	D	57	8	26	9
Aus.10y	BONDS	D	40	14	28	18
Cad.Bond	BONDS	D	60	6	25	9

Table 6.10: Empirical posterior cluster assignment probability $\pi(z_i = k)$ for $i = \{1, \ldots, n\}$ and for $k = \{A, B, C, D\}$.

6.A Appendix: Background Theory

6.A.1 Extreme Value Theory

The focus of this section is to discuss in more detail how the shape of the returns distributions tails is an important feature to distinguish between different markets. Due to the low frequency of observed extreme events, the fitting of long tailed distribution is particularly difficult in practice. From an investment portfolio point of view, being able to deal with rare extreme market events is crucial to make optimal use of capital while avoiding the risk of default. We review the essential theoretical background and discuss different methods available to efficiently estimate the tail shape parameters for each market.

Extreme value theory allows us to make inference on the shape of the tails of a distribution by focusing on the rare large realisations. The objective is not necessarily to fit accurately the bulk of the distribution but to draw reliable inference about the tails from the few extreme observations recorded so far, for a more detailed review of the method see Coles (2001) and Zivot and Wang (2006).

Assuming that $\{X_1, X_2, \ldots\}$ is a collection of i.i.d. random variables each with unknown cumulative distribution function (CDF) $F(x) = P\{X_i \leq x\}$, then we can say that the maximum $M_n = max(X_1, \ldots, X_n)$ has probability

$$P\{M_n \leq x\} = P\{X_1 \leq x, \ldots, X_n \leq x\} = \prod_{i=1}^{n} F(x) = F^n(x)$$

which implies that asymptotically $F^n(x)$ can only converge to 0 or 1 as $n \to \infty$. As a special case, if we were able to find the upper end point x_+ of F we would know that $\forall x < x_+$ then $F^n(x) \to 0$ as $n \to \infty$ and the distribution of M_n degenerates to a point mass on x_+.

If instead we linearly normalize the variable M_n.

$$Z_n = \frac{M_n - a_n}{b_n}$$

where a_n and b_n are sequences of real numbers that control the location and scale

of Z_n as n increases, the distribution of the renormalised variable Z_n can only converge to

$$P\{\frac{M_n - a_n}{b_n} \leq z\} \to H(z) \quad \text{as} \quad n \to \infty$$

where

$$H_{\xi,\mu,\sigma}(x) = \begin{cases} \exp(-(1+\xi(\frac{x-\mu}{\sigma})^{-1/\xi})) & \xi \neq 0, \quad 1+\xi(x-\mu)/\sigma > 0 \\ \exp(-(\exp(-\frac{x-\mu}{\sigma}))) & \xi = 0, \quad x \in \mathbb{R} \end{cases}$$

and μ is the location parameter, σ is the scale parameter and ξ is the shape parameter that determines the tail behaviour of H_ξ while $1/\xi = \alpha$ is called the tail index when $\xi > 0$.

The parameter ξ is what we need to infer from the observed data. It specifies the shape of the distribution at the extremes and controls how fast the tails declines. The three possible cases are:

- **Gumbel:** $\xi = 0$. F is thin tailed, tail declines exponentially.

$$H_{0,0,1}(x) = \exp(-\exp -(x)) \quad \text{for} \quad x \in \mathbb{R}$$

- **Frechet:** $\xi > 0$. F is fat tailed, tail declines by a power function:$1 - F(x) = x^{-1/\xi}L(x)$, for some slowly varying function $L(x)$.

$$H_{\frac{1}{\alpha},0,1}(\alpha \cdot (x-1)) = H_{\frac{1}{\alpha},1,\frac{1}{\alpha}}(x) = \begin{cases} 0 & \text{for} \quad x \leq 0 \\ \exp(-x^{-\alpha}) & \text{for} \quad x > 0 \end{cases}$$

  Unfortunately not all moments are finite for this type of distribution. In fact $E(X^k) = \infty$ for $k \geq \alpha = 1/\xi$

- **Weibull:** $\xi < 0$. In this case the tail of F is finite and all his moments exist.

$$H_{-\frac{1}{\alpha},0,1}(\alpha \cdot (x+1)) = H_{-\frac{1}{\alpha},-1,\frac{1}{\alpha}}(x) = \begin{cases} \exp(-(-x)^{\alpha}) & \text{for} \quad x \leq 0 \\ 1 & \text{for} \quad x > 0 \end{cases}$$

Excess Distribution Function

If we have more information than just the knowledge of extreme values, we can make inference on the tail of the distribution by using all samples above a certain threshold.

We derive the excess distribution function over the threshold u by the conditional probability:

$$F_u(y) = P\{X - u \leq y | X > u\} = \frac{F(y+u) - F(u)}{1 - F(u)}, \quad y > 0$$

While the expected mean function of the excedeances X_i above u is:

$$e(u) = E[X - u | X > u]$$

Generalised Pareto Distribution

For a large enough threshold u, the excess distribution function $F_u(y)$ converges to the Generalised Pareto Distribution $G_{\xi,\mu,\sigma}(y)$

$$G_{\xi,\mu,\tilde{\sigma}}(y) = \begin{cases} 1 - \left(1 + \frac{\xi y}{\tilde{\sigma}}\right)^{-1/\xi} & \text{for} \quad \xi \neq 0, \quad 1 + \xi(x-\mu)/\tilde{\sigma} > 0 \\ 1 - \exp\left(-\frac{y}{\tilde{\sigma}}\right) & \text{for} \quad \xi = 0 \end{cases}$$

where $y > 0$ and $\tilde{\sigma} = \sigma + \xi(u - \mu)$. Note also that while for $\xi < 1$, since for $k \geq \alpha = 1/\xi$ the k-th moment $E[X^k]$ is infinite, the expected mean excess is

$$E[X - u_0 | X > u_0] = \frac{\sigma_{u_0}}{1 - \xi}$$

which can be extended to $u > u_0$ for an appropriate change of scale σ_u:

$$e(u) = E[X - u | X > u] = \frac{\sigma_{u_0} + \xi u}{1 - \xi}$$

6.A.2 Theoretical Random Processes

We review some of the most commonly adopted theoretical models for price and returns series.

Brownian Motion

Brownian motion has been used to model a wide range of phenomena. Its simple formulation makes it a favourite also in finance where it is used to describe the price as a sum of random returns like in the Wold's decomposition, see Feder (1988) and Reif (1965).

We shall start from its simplest version: the one dimensional random walk. Let us consider in the binomial case x as a particle that can move at every step by a fixed length l with a probability p upwards and with probability $q = 1 - p$ down. The mean displacement per step is then given by:

$$\bar{s} = pl + q(-l) = (p - q)l = (2p - 1)l$$

with a dispersion:

$$\overline{(\Delta s)^2} = \overline{s^2} - \bar{s}^2 = l^2[1 - (2p - 1)^2] = 4pql^2$$

After N steps the particle x will be $x = s_1 + s_2 + \cdots + s_N = \sum_{i=1}^{N} s_i$ for which the following is true:

$$\bar{x} = (p - q)Nl$$

$$\overline{(\Delta x)^2} = 4pqNl^2$$

In a slightly more realistic model, if we think that the price Y can move every τ seconds by making a positive or negative jump ξ whose length is set accordingly to the Gaussian distribution:

$$p(\xi, \tau) = \frac{1}{\sqrt{4\pi\mathcal{D}\tau}} \exp\left(-\frac{\xi^2}{4D\tau}\right)$$

where D is the diffusion coefficient:

$$D = \frac{1}{2\tau} E(\xi^2)$$

and $E(\xi^2)$ is the mean square jump distance and also the variance of the sequence of the steps $\{\xi_i\}$:

$$E(\xi^2) = \int_{-\infty}^{\infty} \xi^2 p(\xi, \tau) d\xi = 2D\tau$$

The position of the price Y is then a random function of time $Y(t = n\tau) = \sum_{i=1}^{n} \xi_i$ where the increments distribute according to

$$Y_t - Y_{t_0} \sim \xi |t - t_0|^{1/2}$$

and ξ is a normalised independent Gaussian random process.

Scaling Property of the Brownian Motion

It is worth noting that the Brownian motion has also a nice scaling property. In fact the distribution of ξ for $t = b\tau$ is:

$$p(\xi, b\tau) = \frac{1}{\sqrt{4\pi D b\tau}} \exp\left(-\frac{\xi^2}{4Db\tau}\right)$$

with variance:

$$E(\xi^2) = 2Db\tau = 2Dt$$

For the Brownian motion this mean that whereas the mean displacement of the price $Y(t)$ still has mean:

$$E(Y(t) - Y(t_0)) = 0$$

the variance grows proportionally with time:

$$E([Y(t) - Y(t_0)]^2) = 2D|t - t_0|$$

The transformation that changes the timescale by b and the length scale by $b^{1/2}$

make the distribution invariant:

$$p(b^{1/2}\xi, b\tau) = -b^{1/2}p(\xi, \tau)$$

This property allows us to define the variable y as

$$y = \frac{Y(t) - Y(t_0)}{\sqrt{2D\tau}(|t - t_0|/\tau)^{1/2}}$$

which has standard Gaussian distribution with zero mean and unit variance.

Fractional Brownian Motion

The fractional version of the Brownian motion seem more accurate in describing financial events for only a small increase in complexity, Feder (1988), it only has one more degree of freedom, the exponent H.

The position of the price Y is still a random function of time $Y(t)$ for $t = n\tau$, but the distribution of the increments are now proportional to the time interval raised to the power of H for $0 < H < 1$:

$$Y_t - Y_{t_0} \sim \xi|t - t_0|^H$$

with ξ a normalised independent Gaussian random process as before. The increments have still expected mean

$$E(Y_H(t) - Y_H(t_0)) = 0$$

but the variance is now

$$E([Y(t) - Y(t_0)]^2) = 2D\tau(|(t - t_0)/\tau|)^{2H} \sim |t - t_0|^{2H}$$

and the diffusion coefficient becomes

$$D_H = \frac{1}{2}\frac{d}{dt}E(Y(t)^2) = D|t|^{2H-1}.$$

As for the scaling property, the Fractional Brownian Motion transformation is proportional to $|\Delta t|^H$:

$$Y_H(bt) - Y_H(0) = b^H\{Y_H(bt) - Y_H(0)\} = |\Delta t|^H\{Y_H(1) - Y_H(0)\} \sim |\Delta t|^H$$

In the discrete case we can approximate the process $Y_H(t)$, or the position of the particle after time t, by the summation:

$$Y_H(t) \simeq \frac{1}{\Gamma(H+\frac{1}{2})} \sum_{i=-\infty}^{nt} \left(t - \frac{i}{n}\right)^{H-1/2} \frac{\xi_i}{\sqrt{n}}$$

In the continuous case, instead, the total displacement is obtained by integrating over all previous increments $dB(t')$ of an ordinary Gaussian random process $B(t)$ with average zero and unit variance:

$$Y_H(t) - Y_H(0) = \frac{1}{\Gamma(H+\frac{1}{2})} \int_{-\infty}^{t} K(t-t')dB(t')$$

with

$$K(t-t') = \begin{cases} (t-t')^{H-1/2} & 0 \le t' \le t \\ \{(t-t')^{H-1/2} - (-t')^{H-1/2}\} & t' \le 0 \end{cases}$$

and $\Gamma(x)$ the gamma function.

Chapter 7

Conclusions

The two real life problems we set out to investigate have motivated us to search for a probabilistic model that can simultaneously perform robust clustering and variable selection. We reviewed the existing literature and found scope to develop two new models that are expected to better respond to our requirements and to be robust to the noise and outliers that is generally found in real data.

We have explored both an unsupervised and a supervised route which lead us to propose a penalised mixture of Student's t distributions and mixture of Lasso regressions with t-errors.

We first introduced the penalised mixture of Student's t distributions and illustrated the properties of the model that make it particularly flexible to fit high dimensional data and robust to outliers. Variable selection is performed by imposing an adaptive L_1-norm penalty function acting on the location and the dispersion parameter. To gauge more information about the relative importance of each variable we have also proposed a data resampling procedure which allows us to rank the features and improves on the selection of the true number of clusters. In order to efficiently fit the the model to the data we have derived a modified EM algorithm that returns the maximum likelihood estimates of the unknown mixture parameters.

We have developed the second model in a Bayesian framework and proposed a hierarchical representation of the mixture of regressions that demonstrates the

desired properties of accuracy, robustness and sparsity. Robustness is achieved by allowing the regression errors to be t distributed and adopting convenient priors that induce a Lasso type estimates of the regression coefficients. Sparsity is achieved by assuming a cluster specific binary vector that dictates which variables should be included and which variables should be excluded from the model. In order to estimate the relevant parameters we have implemented a PMCMC algorithm with a Metropolised Gibbs sampler that allows us to approximate the posterior distributions of interest.

The performance of the proposed methods is assessed by generating multiple simulated datasets under different scenarios and illustrating in which situations the models are expected to show a more accurate fit than standard methods.

The algorithms have then been applied for the analysis of two real life problems from bioinformatics and finance respectively, that is identifying clinically distinct subtypes of breast cancer and clustering financial markets based only on some measurable features of their price dynamics. From the analysis of the results we find significant evidence that the inference we draw from the models provide a truthful insight in the problems under investigation.

The three cancer subtypes we isolate are characterized by distinct gene expression profiles and independently confirm the results of other studies on this subject. There is also significant evidence that the genes we select are important to explain the differences we observe in the marginal distribution of some relevant clinical variables and the different prognosis of each cluster.

In clustering financial markets we find that the partition we derive does correspond to groups of markets where the performance of the systematic trading strategy is different. This result suggests that the features we select are in fact informative and could be relevant to design a better trading algorithm.

Future Work

Although it does not seem to have hindered the clustering and variable selection performance of the model, one of the strongest assumptions we made is that

variables are independent. Fitting a full covariance matrix requires a significant increase in complexity and computational effort, but it would lead to a fairer representation of real life situations. Alternatively, we could transpose the data matrix and apply the clustering process also to the variables in order to get more indications, beyond what we can gauge from the the simple correlation, whether there are groups of variables that should be considered as a unity by the model.

In the context of the penalised t mixture model, we believe it would be interesting to explore different types of penalty functions. A pairwise penalisation that acts also on the degrees of freedom parameter seems the most promising route in this area.

In the context of the Bayesian mixture of regressions, there are several aspects which could motivate future work. Firstly, with regards to the theoretical properties of the model. We did not investigate, for example, the issue of Lindley's paradox, which can manifest itself in mixtures (e.g. Jennison (1997)). That is, we would like to know if there are some combination of prior parameters, which would lead one to favouring statistical models with a single component. In connection to this, whether the complex posterior also satisfies a collection of inequalities for model probabilities as is the case for some standard Bayesian mixtures; see Nobile (2005). Secondly, is the actual computational procedure of selecting the number of components. There are at least two options which we intend to consider in future work. The first is simply to use our PMCMC algorithm in each model. Then, as one can easily obtain a marginal likelihood estimate (indeed using the proposed particles - 'all the samples' - see Andrieu et al. (2010)) and compute Bayes factors - see e.g. Nobile (1994). The second idea is to build a trans-dimensional sampler based upon PMCMC and SMC samplers (Del Moral et al., 2006). Here, one uses a trans-dimensional version of the PMMH sampler. Suppose one has a target density $\pi_k(x)$ in dimension k and our overall target density is:

$$\pi(k, x) \propto \pi_k(x)p(k) \quad x \in \mathcal{X}^k \qquad k \in \{1, \ldots, k_{\max}\} = \mathcal{K}$$

where $p(k)$ is a prior on the dimension (here the number of components in the

mixture). Thus we have defined a target density on

$$\bigcup_{k\in\mathcal{K}} \{k\} \times \mathcal{X}^k.$$

Now introduce a sequence of targets of dimension k:

$$\pi_{k,n}(x) \propto \pi_k(x)^{\gamma_n}$$

where $0 < \gamma_1 < \cdots < \gamma_p$ for some $p \geq 1$ given. Our trans-dimensional proposal is as follows: Given a model order k proposal a model order k' and use an SMC sampler to simulate the sequence $\pi_{k',n}$. The acceptance probability of such a move is:

$$1 \wedge \frac{\prod_{n=1}^{p} \frac{1}{N}\sum_{i=1}^{N} w_{n,k'}^i}{\prod_{n=1}^{p} \frac{1}{N}\sum_{i=1}^{N} w_{n,k}^i} \frac{p(k')q(k|k')}{p(k)q(k'|k)}$$

where $q(k'|k)$ is the proposal density of moving from k to k' and

$$\prod_{n=1}^{p} \frac{1}{N}\sum_{i=1}^{N} w_{n,k'}^i$$

is the marginal likelihood estimate from the SMC sampler in dimension k'. This allows one a possibility of producing very competitive trans-dimensional proposals.

Notation

Symbol	Distribution
$\mathcal{B}e$	Bernoulli distribution
$\mathcal{D}ir$	Dirichlet distribution
δ	Dirac delta mass distribution
$\mathcal{IG}a$	Inverse-Gamma distribution
$\mathcal{IW}$	Inverse Wishart distribution
$\mathcal{G}a$	Gamma distribution
$\mathcal{N}$	Gaussian distribution
$\mathcal{M}$	Multinomial distribution
$\mathcal{S}t$	Student's t distribution

Table 7.1: List of notations used for standard distributions.

References

Abba, M. C., Hu, Y., Sun, H., Drake, J. A., Gaddis, S., Baggerly, K., Sahin, A., and Aldaz, C. M. (2005). Gene expression signature of estrogen receptor α status in breast cancer. *BMC Genomics*, 6(4):37.

Aitkin, M. (2001). Likelihood and Bayesian analysis of mixtures. *Statistical Modelling*, 1(4):287–304.

A.K. Dunbier, H. Anderson, Z. Ghazoui, E. Lopez-Knowles, S. Pancholi, R. Ribas, S. Drury, K. Sidhu, A. Leary, L. Martin, M. D. (2011). ESR1 Is Co-Expressed with Closely Adjacent Uncharacterised Genes Spanning a Breast Cancer Susceptibility Locus at 6q25.1. *PLoS Genet*, 7.

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In Petrov, B. N. and Csaki, F., editors, *Second International Symposium on Information Theory*, volume 1 of *Second International Symposium on Information Theory (Tsahkadsor, 1971)*, pages 267–281. Akademia Kiado, Budapest, Akademiai Kiado.

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342.

Arnaud, E. and Le Gland, F. (2009). SMC WITH ADAPTIVE RESAMPLING : LARGE SAMPLE ASYMPTOTICS. In *2009 IEEE Workshop on Statistical Signal Processing*, volume 0, pages 481–484, Cardiff.

Baek, J. and McLachlan, G. J. (2011). Mixtures of common t-factor analyzers for clustering high-dimensional microarray data. *Bioinformatics (Oxford, England)*, 27(9):1269–1276.

Banfield, J. D. and Raftery, A. E. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49(3):803–821.

Barabási, A. L. and Vicsek, T. (1991). Multifractality of self-affine fractals.

Bengtsson, T., Bickel, P., and Li, B. (2008). Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. *Probability and Statistics*, 2:316–334.

Bickel, P., Li, B., and Bengtsson, T. (2008). Sharp failure rates for the bootstrap particle filter in high dimensions. *Pushing the Limits of Contemporary Statistics Contributions in Honor of Jayanta K Ghosh*, 3:318–329.

Bishop, C. M. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1st ed. 20 edition.

Blenkiron, C., Goldstein, L. D., Thorne, N. P., Spiteri, I., Chin, S.-F., Dunning, M. J., Barbosa-Morais, N. L., Teschendorff, A. E., Green, A. R., Ellis, I. O., Tavaré, S., Caldas, C., and Miska, E. A. (2007). MicroRNA expression profiling of human breast cancer identifies new markers of tumor subtype. *Genome Biology*, 8(10).

Box, G. E. P. and Pierce, D. A. (1970). Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models. *Journal of the American Statistical Association*, 65(332):1509–1526.

Callagy, G. M., Pharoah, P. D., Pinder, S. E., Hsu, F. D., Nielsen, T. O., Ragaz, J., Ellis, I. O., Huntsman, D., and Caldas, C. (2006). Bcl-2 is a prognostic marker in breast cancer independently of the Nottingham Prognostic Index. *Clinical Cancer Research*, 12(8):2468–2475.

Calza, S., Hall, P., Auer, G., Björhle, J., Klaar, S., Kronenwett, U., Liu, E. T., Miller, L., Ploner, A., Smeds, J., Bergh, J., and Pawitan, Y. (2006). Intrinsic molecular signature of breast cancer in a population-based cohort of 412 patients. *Breast Cancer Research*, 8(4):R34.

Campbell, J., Lo, A., and Mackinlay, C. (1997). *The Econometrics of Financial Markets, Campbell, Lo, Mackinlay.*

Celeux, G., Hurn, M., and Robert, C. P. (2000). Computational and Inferential Difficulties with Mixture Posterior Distributions. *Journal of the American Statistical Association*, 95(451):957–970.

Coles, S. (2001). *An introduction to statistical modeling of extreme values*, volume 97. Springer.

Day, N. E. (1969). Estimating the components of a mixture of normal distributions.

Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 68(3):411–436.

Del Moral, P., Doucet, A., and Jasra, A. (2011). On Adaptive Resampling Strategies for Sequential Monte Carlo Methods. *Bernoulli (to appear).*

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B Methodological*, 39(1):1–38.

Di Matteo, T. (2007). Multi-scaling in finance. *Quantitative Finance*, 7(1):21–36.

Di Matteo, T., Aste, T., and Dacorogna, M. M. (2004). Long term memories of developed and emerging markets: using the scaling analysis to characterize their stage of development. *Journal of Banking & Finance*, 29(4):46.

Diebolt, J. and Robert, C. P. (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society Series B Methodological*, 56(2):363–375.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, pages 197–208.

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 95(25):14863–14868.

Escobar, M. D. and West, M. (1995). Bayesian Density Estimation and Inference Using Mixtures. *Journal of the American Statistical Association*, 90(430):577–588.

Everitt, B. S. and Hand, D. J. (1981). *Finite Mixture Distributions*, volume 2. London: Chapman and Hall.

Fahrmeir, L., Kneib, T., and Konrath, S. (2009). Bayesian regularisation in structured additive regression: a unifying perspective on shrinkage, smoothing and predictor selection. *Statistics and Computing*, 20(2):203–219.

Fan, J. and Li, R. (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association*, 96(456):1348–1360.

Fearnhead, P. and Meligkotsidou, L. (2007). Filtering Methods for Mixture Models . *Journal Of Computational And Graphical Statistics*, 16(3):586–607.

Feder, J. (1988). *Fractals. Physics of solids and liquids.* Plenum Press.

Fernandez, C. and Steel, M. (1999). Multivariate Student-t regression models: Pitfalls and inference. *Biometrika*, 86(1):153–167.

Figueiredo, M. A. F. and Jain, A. K. (2002). Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396.

Figueiredo, M. A. T. (2000). On Gaussian radial basis function approximations: interpretation, extensions, and learning strategies. *Proceedings 15th International Conference on Pattern Recognition ICPR2000*, pages 618–621.

Fodor, I. K. (2002). A survey of dimension reduction techniques. *Library*, 18(1):1–18.

Fowlkes, E., Gnanadesikan, R., and Kettenring, J. (1988). Variable selection in clustering. *Journal of Classification*, 5(2):205–228.

Fraiman, R., Justel, A., and Svarc, M. (2006). Selection of variables for cluster analysis and classification rules. *Journal of the American Statistical Association*, 103(483):28.

Friedman, J. H. (1991). Multivariate adaptive regression splines. *Annals of Statistics*, 19(1):1–141.

Friedman, J. H. and Meulman, J. J. (2004). Clustering objects on subsets of attributes (with discussion). *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 66(4):815–849.

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-Based Approaches to Calculating Marginal Densities. *Journal of the American Statistical Association*, 85(410):398–409.

George, E. I. and Mcculloch, R. E. (1997). Approaches for Bayesian variable selection. *Statistica Sinica*, 7(2):339–373.

Geweke, J. (1989). Bayesian Inference in Econometric Models Using Monte Carlo Integration. *Econometrica*, 57(6):1317–1339.

Geweke, J. and Porter-Hudak, S. (1983). The estimation and application of long memory time series models. *Journal of Time Series Analysis*, 4(4):221–238.

Ghosh, D. and Chinnaiyan, A. M. (2002). Mixture modelling of gene expression data from microarray experiments. *Bioinformatics*, 18(2):275–86.

Gnanadesikan, R., Kettenring, J. R., and Tsao, S. L. (1995). Weighting and selection of variables for cluster analysis. *Journal of Classification*, 12(1):113–136.

Gneiting, T. and Schlather, M. (2001). Stochastic models which separate fractal dimension and Hurst effect. *Environmental Protection*, 46(069):8.

Goldfeld, S. and Quandt, R. E. (1973). A markov model for switching regression. *Journal of Econometrics*, 1:3–15.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D., and Lander, E. S. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537.

Gopikrishnan, P., Plerou, V., Amaral, L. A. N., Meyer, M., and Stanley, H. E. (1999). Scaling of the distribution of fluctuations of financial market indices. *Physical Review E Statistical Physics Plasmas Fluids And Related Interdisciplinary Topics*, 60(5 Pt A):5305–5316.

Grira, N., Crucianu, M., and Boujemaa, N. (2004). Unsupervised and semi-supervised clustering: a brief survey. *Machine Learning*, pages 1–12.

Guo, J., Levina, E., Michailidis, G., and Zhu, J. (2010). Pairwise variable selection for high-dimensional model-based clustering. *Biometrics*, 66(3):793–804.

Hamilton, J. D. (1994). *Time Series Analysis*, volume 11 of *Springer Texts in Statistics*. Princeton University Press.

Hans, C. (2009). Model uncertainty and variable selection in Bayesian lasso regression. *Statistics and Computing*, 20(2):221–229.

Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models*, volume 43 of *Monographs on Statistics and Applied Probability*. Chapman and Hall.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109.

He, Y., Pan, W., and Lin, J. (2006). Cluster analysis using multivariate normal mixture models to detect differential gene expression with microarray data. *Computational Statistics & Data Analysis*, 51(2):641–658.

Hill, B. M. (1975). A Simple General Approach to Inference About the Tail of a Distribution. *Annals of Statistics*, 3(5):1163–1174.

Hoff, P. D. (2006). Model-based subspace clustering. *Bayesian Analysis*, 1(2):321–344.

Hollander, M. and Wolfe, D. A. (1999). *Nonparametric statistical methods*, volume 2. Wiley-Interscience.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218.

Hurn, M., Justel, A., and Robert, C. P. (2010). Estimating mixtures of regressions. *Journal Of Computational And Graphical Statistics*, 12(1):55–79.

Hurst, H. E. (1951). Long Term Storage Capacity of Reservoirs. *Trans Am Soc Civil Eng*, 116:770–808.

Hurtado, A., Holmes, K. A., Ross-Innes, C. S., Schmidt, D., and Carroll, J. S. (2011). FOXA1 is a key determinant of estrogen receptor function and endocrine response. *Nature Genetics*, 43(1):27–33.

Jacod, J. and Protter, P. (2004). *Probability Essentials*, volume 84. Springer.

Jasra, A., Holmes, C. C., and Stephens, D. A. (2005). Markov Chain Monte Carlo Methods and the Label Switching Problem in Bayesian Mixture Modeling. *Statistical Science*, 20(1):50–67.

Jennison, C. (1997). On Bayesian analysis of mixtures with an unknown number of components - Discussion. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 59(4):731–792.

Jiao, S. (2010). A mixture model based approach for estimating the FDR in replicated microarray data. *Journal of Biomedical Science and Engineering*, 03(03):317–321.

Jiao, S. and Zhang, S. (2008). The t-mixture model approach for detecting differentially expressed genes in microarrays. *Functional & integrative genomics*, 8(3):181–6.

Karlis, D. and Santourian, A. (2009). Model-based clustering with non-elliptically contoured distributions. *Statistics and Computing*, 19(1):73–83.

Kaufman, L. and Rousseeuw, P. J. (2005). *Finding Groups in Data: An Introduction to Cluster Analysis (Wiley Series in Probability and Statistics)*. Wiley-Interscience.

Khalili, A. and Chen, J. (2007). Variable Selection in Finite Mixture of Regression Models. *Journal of the American Statistical Association*, 102(479):1025–1038.

Kim, S., Tadesse, M. G., and Vannucci, M. (2006). Variable selection in clustering via Dirichlet process mixture models. *Biometrika*, 93(4):877–893.

Kotz, S. and Nadarajah, S. (2004). *Multivariate T-Distributions and Their Applications*. Cambridge University Press.

Law, M. H. C., Figueiredo, M. A. T., and Jain, A. K. (2004). Simultaneous feature selection and clustering using mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1154–66.

Lee, A., Caron, F., Doucet, A., and Holmes, C. (2010). A Hierarchical Bayesian Framework for Constructing Sparsity-inducing Priors. *Computing*, pages 1–18.

Lin, T. I., Lee, J. C., and Yen, S. Y. (2007). Finite mixture modelling using the skew normal distribution. *Statistica Sinica*, 17:909–927.

Liu, C. and Rubin, D. B. (1995). ML estimation of the t distribution using EM and its extensions, ECM and ECME. *Statistica Sinica*, 5(1):19–39.

Liu, J. S. (2001). *Monte Carlo Strategies in Scientific Computing*. Springer Series in Statistics. Springer.

Liu, J. S., Zhang, J. L., Palumbo, M. J., and Lawrence, C. E. (2003). Bayesian Clustering with Variable and Transformation Selections. *Statistics*, 7(2001):249–275.

Liu, X. and Rattray, M. (2010). Including probe-level measurement error in robust mixture clustering of replicated microarray gene expression. *Statistical Applications in Genetics and Molecular*, 9(1).

Ljung, G. M. and Box, G. E. P. (1978). On a measure of lack of fit in time series models. *Biometrika*, 65(2):297–303.

Lo, A. W. and MacKinlay, A. C. (1988). Stock market prices do not follow random walks: evidence from a simple specification test. *Review of Financial Studies*, 1(1):41–66.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In Cam, L. M. L. and Neyman, J., editors, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. California, USA, University of California Press.

Mantegna, R. N. and Stanley, E. H. (1999). *An Introduction to Econophysics: Correlations and Complexity in Finance*. Cambridge University Press.

Marin, J., Mengersen, K., and Robert, C. (2005). Bayesian Modelling and Inference on Mixtures of Distributions. *Handbook of Statistics*, 25(05):459–507.

Marshall, A. (1956). The Use of Multi-Stage Sampling Schemes in Monte Carlo Computation. In Meyer, M., editor, *Symposium on Monte Carlo Methods*, pages 123–140.

Maugis, C., Celeux, G., and Martin-Magniette, M. L. (2009). Variable selection in model-based clustering: A general variable role modeling. *Computational Statistics & Data Analysis*, 53(11):3872–3882.

McLachlan, G. and Peel, D. (2000). *Finite Mixture Models (Wiley Series in Probability and Statistics)*. Wiley-Interscience.

McLachlan, G. J., Bean, R. W., and Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18(3):413–422.

McLachlan, G. J. and Krishnan, T. (2008). *The EM Algorithm and Extensions (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2 edition.

Meinshausen, N. and Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 72(4):417–473.

Melnykov, V. and Maitra, R. (2010). Finite mixture models and model-based clustering. *Statistics Surveys*, 4:80–116.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equations of state calculations by fast computing machine. *Journal of Chemical Physics*, 21:1087–1091.

Meyn, S. P. and Tweedie, R. L. (1993). Markov Chains and Stochastic Stability. *Journal of the American Statistical Association*, 92(438):792.

Mukhopadhyay, S. and Bhattacharya, S. (2011). Perfect Simulation for Mixtures with Known and Unknown Number of components. page 39.

Naderi, A., Teschendorff, A. E., Barbosa-Morais, N. L., Pinder, S. E., Green, A. R., Powe, D. G., Robertson, J. F. R., Aparicio, S., Ellis, I. O., Brenton, J. D., and

Caldas, C. (2007). A gene-expression signature to predict survival in breast cancer across independent data sets. *Oncogene*, 26(10):1507–16.

Nelder, J. A. and Wedderburn, R. W. M. (1972). Generalized Linear Models. *Journal Of The Royal Statistical Society Series A General*, 135(3):370–384.

Newcomb, S. (1886). A Generalized Theory of the Combination of Observations So As To Obtain the Best Results. *American Journal of Mathematics*, 8:343–366.

Newey, W. K. and West, K. D. (1987). A Simple, Positive Semidefinite, Heteroskedasticity and Autocorrelation Consistent Covariance-Matrix. *Econometrica*, 55(3):703–708.

Nicolau, M., Levine, A. J., and Carlsson, G. (2011). Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proceedings of the National Academy of Sciences of the United States of America*, 108(17):7265–70.

Nobile, A. (1994). *Bayesian Analysis of Finite Mixture Distributions*. PhD thesis, Carnegie Mellon University.

Nobile, A. (2005). On the posterior distribution of the number of components in a finite mixture. *Annals of Statistics*, 32(5):2044–2073.

Pan, W. and Shen, X. (2007). Penalized Model-Based Clustering with Application to Variable Selection. *Journal of Machine Learning Research*, 8:1145–1164.

Park, T. and Casella, G. (2008). The Bayesian Lasso. *Journal of the American Statistical Association*, 103(482):681–686.

Parsons, L., Haque, E., and Liu, H. (2004). Subspace clustering for high dimensional data: a review. *SIGKDD Explor. Newsl.*, 6(1):90–105.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions A*, 185(A):71–110.

Peel, D. and McLachlan, G. J. (2000). Robust mixture modelling using the t distribution. *Statistics and Computing*, 10(4):339–348.

Peters, E. E. (1991). *Chaos and Order in the Capital Markets*. John Wiley & Sons.

Qu, Y. and Xu, S. (2004). Supervised cluster analysis for microarray data based on multivariate Gaussian mixture. *Bioinformatics (Oxford, England)*, 20(12):1905–13.

Quandt, R. and Ramsey, J. (1978). Estimating Mixtures of Normal Distributions and Switching Regressions. *Journal of the American Statistical Association*, 73(364):730–738.

Raftery, A. E. and Dean, N. (2006). Variable Selection for Model-Based Clustering. *Journal of the American Statistical Association*, 101(473):168–178.

Reif, F. (1965). Fundamentals of Statistical and Thermal Physics (McGraw-Hill Series in Fundamentals of Physics).

Richardson, S. and Green, P. J. (1997). On Bayesian Analysis of Mixtures with an Unknown Number of Components (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(4):731–792.

Robert, C. and Casella, G. (2011). A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data. *Statistical Science*, 26(1):102–115.

Robert, C. P. and Casella, G. (2004). *Monte Carlo Statistical Methods*, volume 96 of *Springer Texts in Statistics*. Springer.

Roberts, G. O., Gelman, A., and Gilks, W. R. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *The Annals of Applied Probability*, 7(1):110–120.

Roberts, S. J., Husmeier, D., Rezek, I., and Penny, W. (1998). Bayesian approaches to Gaussian mixture modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1133–1142.

Roeder, K. and Wasserman, L. (1997). Practical Bayesian Density Estimation using Mixtures of Normals. *Journal of the American Statistical Association*, 92(439):894.

Russell, B. (1912). The Problems of Philosophy. *Philosophy East and West*, 49(1):96.

Schäfer, C. and Chopin, N. (2011). Sequential Monte Carlo on large binary sampling spaces.

Schneider, J., Ruschhaupt, M., Buness, A., Asslaber, M., Regitnig, P., Zatloukal, K., Schippinger, W., Ploner, F., Poustka, A., and Sültmann, H. (2006). Identification and meta-analysis of a small gene expression signature for the diagnosis of estrogen receptor status in invasive ductal breast cancer. *International journal of cancer Journal international du cancer*, 119(12):2974–2979.

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

Sharpe, W. F. (1966). Mutual Fund Performance. *The Journal of Business*, 39(1):119–138.

Smolkin, M. and Ghosh, D. (2003). Cluster stability scores for microarray data in cancer studies. *BMC Bioinformatics*, 4(1):36.

Sorlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H., Hastie, T., Eisen, M. B., Van De Rijn, M., Jeffrey, S. S., Thorsen, T., Quist, H., Matese, J. C., Brown, P. O., Botstein, D., Lø nning, P. E., and Bø rresen Dale, A.-L. (2001). Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences of the United States of America*, 98(19):10869–10874.

Sorlie, T., Tibshirani, R., Parker, J., Hastie, T., Marron, J. S., Nobel, A., Deng, S., Johnsen, H., Pesich, R., Geisler, S., Demeter, J., Perou, C. M., Lø nning, P. E., Brown, P. O., Bø rresen Dale, A.-L., and Botstein, D. (2003). Repeated

observation of breast tumor subtypes in independent gene expression data sets. *Proceedings of the National Academy of Sciences of the United States of America*, 100(14):8418–8423.

Sotiriou, C. and Pusztai, L. (2009). Gene-expression signatures in breast cancer. *The New England Journal of Medicine*, 360(8):790–800.

Sprott, J. C. (2003). *Chaos and Time-Series Analysis*. Oxford University Press.

Steinley, D. and Brusco, M. J. (2008). Selection of Variables in Cluster Analysis: An Empirical Comparison of Eight Procedures. *Psychometrika*, 73(1):125–144.

Tadesse, M. G., Sha, N., and Vannucci, M. (2005). Bayesian Variable Selection in Clustering High-Dimensional Data. *Journal of the American Statistical Association*, 100(470):602–617.

Thakkar, A. D., Raj, H., Chakrabarti, D., Ravishankar, N., Saravanan, N., Muthuvelan, B., Balakrishnan, A., and Padigaru, M. (2010). Identification of Gene Expression Signature in Estrogen Receptor Positive Breast Carcinoma. *Biomarkers in Cancer*, 2:1–15.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B Methodological*, 58(1):267–288.

Tipping, M. E. (2001). Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1(3):211–244.

Titterington, D. M., Smith, A. F. M., and Makov, U. E. (1985). *Statistical Analysis of Finite Mixture Distributions*, volume 7. Chichester: Wiley.

Van't Veer, L. J., Dai, H., Van De Vijver, M. J., He, Y. D., Hart, A. A. M., Mao, M., Peterse, H. L., Van Der Kooy, K., Marton, M. J., Witteveen, A. T., Schreiber, G. J., Kerkhoven, R. M., Roberts, C., Linsley, P. S., Bernards, R., and Friend, S. H. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–6.

Wang, S. and Zhu, J. (2008). Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, 64(2):440–448.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244.

Wedel, M. and DeSarbo, W. S. (1995). A mixture likelihood approach for generalized linear models. *Journal of Classification*, 12(1):21–55.

Witten, D. M. and Tibshirani, R. (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490):713–726.

Wolfe, J. H. (1970). Pattern Clustering By Multivariate Mixture Analysis. *Multivariate Behavioral Research*, 5(3):329–350.

Xie, B., Pan, W., and Shen, X. (2008a). Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. *Electronic journal of statistics*, 2:168–212.

Xie, B., Pan, W., and Shen, X. (2008b). Variable Selection in Penalized Model-Based Clustering via Regularization on Grouped Parameters. *Biometrics*, 64(3):921–930.

Yau, C. and Holmes, C. (2011). Hierarchical Bayesian nonparametric mixture models for clustering with variable relevance determination. *Bayesian analysis (Online)*, 6(2):329–352.

Zhu, X. and Goldberg, A. B. (2009). Introduction to Semi-Supervised Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1):1–130.

Zivot, E. and Wang, J. (2006). *Modeling Financial Time Series with S-PLUS*, volume 49. Springer.

Zou, H. (2006). The Adaptive Lasso and its Oracle Properties. *Journal of the American Statistical Association*, 101(476):1418–1429.

"Pars virtutis disciplina constat, pars exercitatione; et discas oportet et quod didicisti agendo confirmes"

Lucius Annaeus Seneca