Bayesian Latent Variable Models with Applications

A thesis presented for the degree of

Doctor of Philosophy of the University of London

and the

Diploma of Imperial College

by

Aidan Michael O'Sullivan

Department of Mathematics

Imperial College

180 Queen's Gate, London SW7 2BZ

DECEMBER 2, 2013

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: Aidan O'Sullivan

Copyright

An lámh fhoisteanach abú

Abstract

The massive increases in computational power that have occurred over the last two decades have contributed to the increasing prevalence of Bayesian reasoning in statistics. The often intractable integrals required as part of the Bayesian approach to inference can be approximated or estimated using intensive sampling or optimisation routines. This has extended the realm of applications beyond simple models for which fully analytic solutions are possible. Latent variable models are ideally suited to this approach as it provides a principled method for resolving one of the more difficult issues associated with this class of models, the question of the appropriate number of latent variables. This thesis explores the use of latent variable models in a number of different settings employing Bayesian methods for inference.

The first strand of this research focusses on the use of a latent variable model to perform simultaneous clustering and latent structure analysis of multivariate data. In this setting the latent variables are of key interest providing information on the number of sub-populations within a heterogeneous data set and also the differences in latent structure that define them. In the second strand latent variable models are used as a tool to study relational or network data. The analysis of this type of data, which describes the interconnections between different entities or nodes, is complicated due to the dependencies between nodes induced by these connections. The conditional independence assumptions of the latent variable framework provide a means of taking these dependencies into account, the nodes are independent conditioned on an associated latent variable. This allows us to perform model based clustering of a network making inference on the number of clusters. Finally the latent variable representation of the network, which captures the structure of the network in a different form, can be studied as part of a latent variable framework for detecting differences

between networks.

Approximation schemes are required as part of the Bayesian approach to model estimation. The two methods that are considered in this thesis are stochastic Markov chain Monte Carlo methods and deterministic variational approximations. Where possible these are extended to incorporate model selection over the number of latent variables and a comparison, the first of its kind in this setting, of their relative performance in unsupervised model selection for a range of different settings is presented. The findings of the study help to ascertain in which settings one method may be preferred to the other.

Acknowledgements

There are a number of people without whom this work would not have been possible. Firstly I would like to thank my supervisors Dr. Iead Rezek and Dr. Niall Adams, their encouragement and guidance has been a tremendous help. I consider myself very fortunate to have been able to draw upon their insight and expertise.

I am also grateful to Dr. Patrick Rubin-Delanchy for all of his assistance in my first year. His knowledge of and enthusiasm for statistics made sharing an office an enjoyable and beneficial experience and I am sure he will make an excellent supervisor someday.

I am grateful to the EPSRC and BAE systems for providing funding for this research.

Finally I would like to thank my parents for the opportunities they have given me throughout my life and the support they have shown at all times during this work.

Aidan Michael O'Sullivan

Table of contents

List of Figures

List of Tables

Chapter 1

Overview

This thesis considers the use of latent variables to model dependency structure in both multivariate and relational data. Latent variables may be used simply as a mathematical convenience, for example they allow us to take advantage of conditional independence properties to handle dependencies inherent in network data, or alternatively the latent variables may have a fundamental meaning which is the target of inference. In this case, the fundamental meaning is not present in the manifest (or observed) variables.

Although a very useful tool, the introduction of latent variables creates a number of problems which must be dealt with such as determining the appropriate dimension of the latent space. These problems themselves can often be of interest, rather than being merely computational inconveniences, particularly when using latent variables to capture cluster structure in the data and determining the appropriate number of clusters. In order to tackle these problems associated with latent variable models we follow the Bayesian approach to inference. A Bayesian approach allows unknown parameters, such as the number of clusters, to be handled in a principled manner, by expressing our uncertainty through prior distributions and inferring posterior distributions from the observed data. The strength of this approach is particularly apparent when it comes to tackling model selection issues, with the Bayesian engine carrying an inherent penalisation for overly complex models.

Although principled and relatively straightforward in its theory the Bayesian approach to inference, in this setting, leads to integrals that are analytically intractable. In order to approximate these integrals sophisticated computational techniques are required. In this

thesis two approximation methods are studied; Markov Chain Monte Carlo based methods capable of trans-dimensional moves and methods based on variational approximations. These are two different solutions to the same problem and they each have have their own strengths and weaknesses. To gain a deeper understanding of the context in which the choice of one algorithm may be more appropriate than the other an empirical comparison of these methods is conducted for a range of different settings. This work constitutes a novel contribution to the model selection literature building on a recent study in [21] and extending this to the unsupervised domain.

Armed with methods to handle the problems associated with latent variables and an understanding of the associated Bayesian model selection methods the latent variable model framework can be used to tackle complex problems without the need for restrictive a priori assumptions. We begin by considering joint inference of both the number of clusters, K, and the dimensionality of the latent space, q, in a Mixture of Factor Analysers (MFA) model. Estimation of q and K are intrinsically linked, for example if the latent space defined is insufficient to separate individual clusters the value of K will be underestimated. We further extend the problem by allowing each cluster to inhabit its own latent space allowing clusters defined by differences in latent structure to be detected. In this setting the number of possible models increases massively and standard model selection methods such as Bayes Factors are infeasible making these more complex model selection methods the only viable solution.

The statistical analysis of autism data is an application where the above approach is particularly useful. Autism is a neural disorder characterised by impaired social interaction and verbal and non-verbal communication skills and restricted or repetitive behaviour, with patients existing on a broad spectrum of disorder. The heterogeneous nature of the population complicates analysis and here we deploy the MFA on behavioural data collected as part of research conducted by the International Molecular Genetic Study of Autism Consortium (IMGSAC). Analysing the data in this manner to detect sub-populations exhibiting different latent structure is a problem of real value that can both uncover the unknown number of sub-populations present and reveal the differences that define these groups. Analysis at this level of flexibility and sophistication has never been performed previously in this setting and represents a novel contribution to the statistical analysis of autism data.

The latent variable approach can be used to model network data in an elegant fashion that respects the unique characteristics of this type of data. Networks are composed of entities, referred to as nodes or vertices, and their connections, referred to as links or edges. The connected nature of networks means that these nodes and edges are inherently dependent on each other and standard statistical methods based on assumptions of independence between samples are inappropriate. A latent variable model provides a means of handling this dependence by introducing a latent variable for each node. This allows us to treat the nodes as independent, conditional on the latent variable. A model based on this principle is the Latent Position model [63] in which the latent variables are positions in Euclidean space. The model has been extended to incorporate model based clustering of a network. This variation on the original model is the Latent Position Cluster Model (LPCM) [58] which will be used extensively throughout this thesis. The form of the likelihood of the model makes Bayesian inference possible and we incorporate MCMC based Bayesian model selection techniques to tackle the problem of inferring the number of communities in a network. This extends the work of the previous groups [63, 58]who developed the models and our contribution is solely from a model estimation and selection perspective, incorporating trans-dimensional MCMC based methods for the first time in this setting.

The analysis of network data suffers from a problem with scale, as the size of the network increases computationally intensive methods such as MCMC quickly become infeasible. Informed by the comparison of variational and MCMC methods in Chapter 3, in which it was shown that comparable performance can be obtained from the variational approach in much shorter timescales provided there is sufficient data, a variational approach to estimating the Latent Position and LPCM models is developed. This allows these latent variable methods to be applied to larger networks.

Finally the latent representation of a network obtained from the LPCM is of interest in itself. It provides a model-based method for visualising the network and also gives us a multivariate representation on which standard multivariate analysis tools such as Canonical Correlation Analysis (CCA) can be deployed. CCA measures the correlations between two sets of data. This allows two networks to be compared, or a single network to be monitored over time to detect changes in structure. A probabilistic formulation of CCA as a latent variable model allows us to construct a test statistic with confidence intervals and

the overall method is a novel approach to detecting changes in a network. We combine this change detection method with the MCMC based model selection techniques applied to the LPCM to study the Enron email corpus. The Enron email corpus is a real-world dataset of email communications from a company during a time of severe crisis. These methods allow us to study the community structure of the network and detect changes in the patterns of communication, which may indicate the occurrence of extreme events such as the resignation of the CEO or filing for bankruptcy. This final application brings practically all elements of this thesis together.

1.1 Contributions

This thesis makes a number of contributions in the following areas:

- **Methods for Bayesian Model Selection:** The comparative study provided in Chapter 3 evaluates the performance of MCMC and variational methods for model selection in a range of different settings, for example varying the number of points, the difficulty of the clustering problem and the dimensionality. A similar study of sorts was undertaken by [21] however our work extends this beyond the regression setting to that of unsupervised model selection. Understanding the different characteristics of both methods is important to allow the most appropriate method to be used for different problems and our results provide strong evidence to suggest that given sufficient amounts of data the variational approach can match the accuracy of MCMC based methods but in time scales that are orders of magnitude shorter.

- **Statistical Analysis of Autism Data:** A large volume of literature exists on the analysis of the Autism Diagnostic Interview - Revised with broadly speaking two objectives, to stratify a cohort of subjects into more homogeneous sub-populations based on specific behavioural traits or to analyse the structure of the interview for possible improvements. The methods deployed on the ADI-R in this thesis are some of the most sophisticated ever used and to the best of our knowledge represent the only fully unsupervised Bayesian model based clustering of the data that also incorporates latent structure analysis. The results obtained show the existence of a number

of sub-populations in a carefully chosen study sample and also differences in latent structure between these groups. This is a further illustration of the heterogeneous nature of the autism population and the need for methods which can take into account the inherent heterogeneity.

- **Network Community Detection:** The Latent Position Cluster Model provides a tool for performing model based clustering of a network. We extend the original MCMC based model estimation method to incorporate a sampler capable of trans-dimensional moves to allow inference over the number of clusters K. This represents an improvement on the original method for selecting between models of differing K by comparing pairs of models using Bayes factors, a weakness which was remarked upon in the discussion of the original RSS read paper [58].

- **Efficient Algorithms for Networks:** A key issue affecting the analysis of networks is the massive computational burden associated with increases in network size. This makes MCMC based estimation infeasible for large networks of more than a thousand nodes. An alternative method of estimation by variational approximations is faster by orders of magnitude and we derive a set of update equations that allow the Latent Position and LPCM to be estimated in this fashion. There has been previous work on this problem [114], highlighting the need for efficient methods of estimation, however the authors were forced to resort to numerical optimisation routines to update the model parameters due to the complexity of the derivations. We present a fully realised probabilistic solution utilising an alternative but equivalent phrasing of the model.

- **Detecting Changes in the Structure of Networks:** There are two main approaches to the problem of detecting differences between or determining changes in the structure of networks. These are feature based [87] or those that work in the graph space, generally using subgraph comparison [67] or entropy measures [27]. The latent variable approach of the LPCM allows us to contribute a unique approach based on analysing an embedding of the network in Euclidean space. This differs to methods that work in the graph domain and is more closely linked to methods that extract

and monitor specific features of a network. Both approaches have their weaknesses, methods in the graph space carry a huge computational burden while the feature based method inherently restricts the type of changes that can be detected to only those that provoke change in those features. The method proposed here uses Canonical Correlation Analysis to compare representations of networks in Euclidean space which we feel is a more principled approach than selecting arbitrary features and less costly than the graph based approaches. A latent variable model formulation of CCA allows a test statistic with confidence intervals to be derived.

1.2 Thesis Structure

The thesis is structured as follows:

- Chapter 2 provides the background for the methods that will be used throughout this thesis. The general framework of latent variable models is described as well as some specific models which will feature prominently. A model-based approach naturally incurs the question of model selection and the Bayesian approach to model selection is discussed. This approach requires the computation of the marginal likelihood which is often an intractable integral and we discuss two approximation methods for estimating this as part of the model selection problem, these are Markov Chain Monte Carlo and variational approximations. The methods discussed here form the core of this thesis and are deployed in Chapter 3.

- In Chapter 3 we deploy the model estimation techniques described in Chapter 2 to the problem of estimating both the number of clusters and the dimensionality of the latent space in a model that combines two of the latent variable models described in Sections 2.1.1 and 2.1.4, Factor Analysis and the Finite Mixture Model. This two tiered model selection problem is made more complex by allowing the dimension of the latent space to vary between clusters and a sophisticated birth-death MCMC method is deployed which is explained in detail. This is compared with the corresponding variational approach in a simulation study in which the performance of both approximation methods is investigated for a range of different settings. Hav-

ing studied these methods extensively we then apply them to the analysis of a data set of behavioural data from a cohort of subjects with autism. The flexibility of the model selection techniques are a key advantage allowing sub-populations within a supposedly homogeneous sample to be detected while also uncovering differences in latent structure between the sub-populations which may be of interest in characterising these groups.

- Chapter 4 marks a shift in the focus of the thesis and from hereon we concentrate on the analysis of network or graph data, but still maintaining a latent variable approach. The chapter begins with a description of some of the characteristics of network data that must be captured by any suitable modelling approach and some basic models from the literature are introduced. These will be useful to illustrate some of the characteristic behaviour of networks and also as a means to generate synthetic data in later chapters. Next the specific model utilised, the Latent Position Cluster Model, which accounts for the dependency between the nodes and edges of a network by conditioning on a latent position variable in Euclidean space, is introduced. These latent positions are assumed to be drawn from a mixture of Gaussian distributions and we incorporate the birth-death MCMC methods presented in Chapter 3 to perform inference on the number of mixtures.

- Chapter 5 is motivated by the results of the simulation study in Chapter 3 where we showed that the performance obtained from the variational method for model estimation can match that of the MCMC approach, given sufficient data. Scale is a key problem with network data and as the size of the network increases the computational burden makes MCMC methods infeasible. In this chapter a variational method for estimating the LPCM based on a alternate phrasing of the model using cosine angle distance is developed. Similar work has been attempted in [114] however the approach developed here does not require the use of numerical optimisation routines and represents the first fully probabilistic solution.

- Chapter 6 tackles the problem of detecting changes in network structure using the latent variable embedding obtained from the LPCM. We review some of the relevant

literature in the area before describing our own particular method which uses Canonical Correlation Analysis to compare embeddings. The probabilistic formulation of CCA as a latent variable model allows us, within the Bayesian framework, to construct a test statistic with confidence intervals based on the posterior distribution of the transformation matrices. The derivation of this test statistic is described in detail and the method is tested extensively in a number of experiments using synthetic data from the simple network models discussed in Chapter 4. These experiments also allow us to characterise the capabilities of the method. We then conclude by applying the method to two data sets: the VAST challenge 2008 data set and the Enron Email corpus. This chapter contains extended versions of material previously published as [99] and is reproduced with permission from the IEEE, ©2012.

- Chapter 7 concludes the thesis with a discussion of some of our findings and avenues of further exploration deriving from the work.

1.3 Glossary

A synopsis of the notation used throughout this report is provided here for reference. In general we will adhere to the conventions defined but it may be necessary at times to reuse variables in different contexts. The variables n, p, q and K refer to the number of data points, the dimensionality of the data points, the number of factors and the number of clusters respectively:

$\boldsymbol{X}$	data matrix with n rows corresponding to instances and p corresponding to variables	$n \times p$
$\boldsymbol{x}$	data vector for a single observation of p variables	$1 \times p$
x_i	scalar observation, $i \in \{1, \ldots, p\}$	1×1
$\boldsymbol{Y}$	generally reserved for the adjacency matrix of a network	$n \times n$
y_{ij}	binary variable used to denote the presence or absence of an edge between two nodes in a network	1 or 0
$\boldsymbol{Z}$	matrix of latent variables with n rows corresponding to instances and q corresponding to independent variables	$n \times q$
$\boldsymbol{\Lambda}$	factor loadings matrix	$p \times q$
$\boldsymbol{\mu}$	mean vector	$1 \times p$
$\boldsymbol{\epsilon}$	noise matrix	$n \times p$
$\boldsymbol{\Psi}$	variance matrix of noise	$p \times p$
$\boldsymbol{\Sigma}$	covariance matrix of data	$p \times p$
$\boldsymbol{\pi}$	vector of mixing proportions	$1 \times K$
$\mathbf{0}_p$	vector of zeros	$1 \times p$
$\boldsymbol{\theta}$	collection of model parameters	
$p(.)$	denotes a probability distribution	
$q(.)$	denotes a 'proposal' probability distribution	
$q^*(.)$	used to indicate the 'optimal' probability distribution	

Chapter 2

Background

Latent variable models will feature extensively throughout this thesis. In this chapter a general overview of the framework of this class of model is provided and a number of specific models which will be used in later chapters are introduced. As discussed in Chapter 1 a Bayesian approach to model estimation is preferred and this approach is extended to incorporate model selection. Bayesian model selection reduces the number of modelling assumptions, allowing key parameters such as the dimensionality of the latent space to be inferred for a fully unsupervised method. The cost of this increased model flexibility is added complexity and the second half of this chapter is dedicated to an exploration of relevant parts of the theory and methodology of Bayesian model selection. The methods considered fall into two categories: MCMC methods and variational approximations. An empirical comparison of both approaches will be performed in Chapter 3 but first we provide a detailed introduction to the theory behind both.

2.1 Latent Variable Models

Many of the fundamental methods of modern statistics, such as the design of experiments, have their roots in the analysis of physical data from the agricultural sciences [45, 46]. In contrast latent variable models have their origins in the study of less tangible data from the social and behavioural sciences [121, 129]. Latent variables can be used to represent many phenomena such as 'true' variables measured with error, unobserved heterogeneity or sim-

ply concepts that do not admit direct measurements [9, 14, 80, 118]. Many concepts in the social and behavioural sciences, such as social class, personality, intelligence and ambition are of this type. In order to obtain information on such variables researchers are required to consider other variables which can be measured (i.e. are *manifest* variables) and which are related to the original quantities of interest, but which may contain additional error or noise. For example we may attempt to measure intelligence by a battery of tests including an I.Q. test, an arithmetic test, a comprehension test and a spelling test. However none of these tests are a pure measure of intelligence, arithmetic may additionally involve numerical ability, spelling will depend to some extent on memory, comprehension on verbal facility and all the tests will be subject to sampling fluctuation as well as unpredictable measurement errors. An individual's intelligence will need to be estimated in some way from that individual's test scores and to do this a model must link the latent and manifest variables [80]. In this thesis we simply consider a latent variable as a random variable whose realisation is hidden from us. This is in contrast to manifest variables where the realisations are observed, we will use this term interchangeably with the term data throughout this thesis.

Latent variable models provide a powerful approach to probabilistic modelling. By defining a joint distribution over manifest and latent variables the corresponding distribution of the observed variables can be obtained by marginalisation [13]:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z} \tag{2.1}$$

where $\boldsymbol{x} = (x_1, \ldots, x_p)$ are the manifest variables and $\boldsymbol{z} = (z_1, \ldots, z_q)$ are the latent variables, with $q < p$. Of course we need a way to relate the latent variables to the manifest variables and this brings us to the basic premise for latent variable models which is that given p correlated manifest variables, $\boldsymbol{x}$, the observed associations among them can be explained by finding q uncorrelated latent variables, $\boldsymbol{z}$, such that the x_i, $i \in \{1, \ldots, p\}$, are conditionally independent given the values of the $\boldsymbol{z}$ [9],

$$p(\boldsymbol{x}) = \int p(\boldsymbol{z}) \prod_{i=1}^{p} p(x_i|\boldsymbol{z})d\boldsymbol{z}. \tag{2.2}$$

Next the conditional distribution $p(\boldsymbol{x}|\boldsymbol{z})$ is expressed in terms of a mapping from latent

variables to data variables so that:

$$\boldsymbol{x} = f(\boldsymbol{z}; \boldsymbol{w}) + \boldsymbol{\epsilon}, \tag{2.3}$$

where $f(\boldsymbol{z}; \boldsymbol{w})$ is a function of the latent variable $\boldsymbol{z}$ with parameters $\boldsymbol{w}$ and $\boldsymbol{\epsilon}$ is an independent noise process. If the components of $\boldsymbol{\epsilon}$ are uncorrelated the conditional distribution for $\boldsymbol{x}$ will factorise as in Equation 2.2.

The definition of the latent variable model is completed by specifying the distribution $p(\boldsymbol{\epsilon})$, the mapping $f(\boldsymbol{z}; \boldsymbol{w})$ and the marginal or prior distribution $p(\boldsymbol{z})$ [80]. Different assumptions lead to different models, for example the well known finite mixture model which will be discussed in Section 2.1.4 is a latent variable model in which the latent variables are discrete component labels. Alternatively if we consider continuous latent variables a number of models are possible. Within this thesis we will restrict ourselves to linear Gaussian models where the mapping from latent variables to manifest variables is assumed to be linear, for example Factor Analysis, see Section 2.1.1, where the latent variables are assumed to follow a Gaussian distribution. It is however also possible to have a non-linear mapping, $f(.)$, from latent variable to data space. While we do not utilise any of these non-linear models, in Section 3.1 we will show how combining mixture models with a linear Gaussian factor model, described in detail in the following section, makes it possible to handle non-linear data.

2.1.1 Factor Analysis

The general framework outlined above permits a vast array of models, in order to progress it is necessary to make assumptions about the distributions of the manifest and latent variables. In the specific case of both continuous manifest and latent variables a model must be formulated that satisfies Equation 2.2. If $\boldsymbol{x}$ and $\boldsymbol{z}$ are assumed to follow multivariate Gaussian distributions then the joint distribution, $p(\boldsymbol{x}, \boldsymbol{z})$, and the conditional distribution $p(\boldsymbol{x}|\boldsymbol{z})$ are also both Gaussian. Moreover the conditional mean of $\boldsymbol{x}$ given $\boldsymbol{z}$ is linear in $\boldsymbol{z}$ and the conditional covariance matrix does not depend on $\boldsymbol{z}$ [80]. From these assumptions

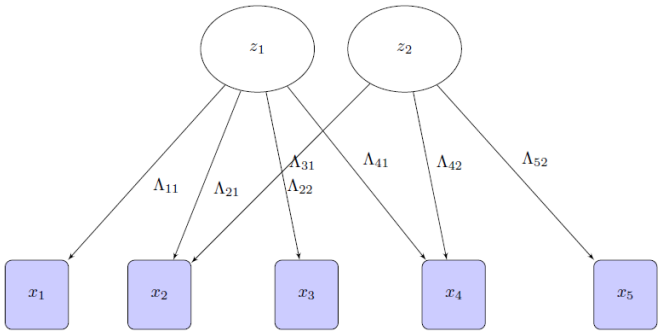


Figure 2.1: Schematic showing how the loadings matrix, $\mathbf{\Lambda}$, encodes the network of dependencies within the manifest variables

the following generative equation is obtained:

$$\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{z}\mathbf{\Lambda}^T + \boldsymbol{\epsilon}. \tag{2.4}$$

The latent variables $\boldsymbol{z}$ are referred to as *factors* and if they are assumed to follow a standard Gaussian distribution, $p(\boldsymbol{z}) \sim \mathcal{N}(\mathbf{0}_q, \boldsymbol{I}_q)$, then we have the *linear Gaussian factor model* [9, 42]. The p-dimensional random vector $\boldsymbol{\epsilon}$ is distributed $\mathcal{N}(\mathbf{0}_p, \mathbf{\Psi})$, where $\mathbf{\Psi}$ is a $p \times p$ diagonal covariance matrix. The diagonality of $\mathbf{\Psi}$ is not so much an assumption of the model but rather a consequence of the conditional independence postulate, Equation 2.2. The manifest variables are independent given the factors [53]. A parameter that will be of key interest throughout this thesis is the loadings matrix, $\mathbf{\Lambda}$, a $p \times q$ matrix which captures all the correlation between the latent and manifest variables as illustrated in Figure 2.1. The mean vector of the manifest variables is captured by the parameter $\boldsymbol{\mu}$ [9].

According to this linear Gaussian model, the marginal distribution of $\boldsymbol{x}$ is:

$$p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Lambda}\mathbf{\Lambda}^T + \mathbf{\Psi}),$$

and the goal of factor analysis (FA) is to find the appropriate values of q, $\mathbf{\Lambda}$ and $\mathbf{\Psi}$ that model the covariance structure of the population [80]. From these assumptions it can be

seen that the conditional distribution of $\boldsymbol{x}$ given $\boldsymbol{z}$ is:

$$p(\boldsymbol{x}|\boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{z}\boldsymbol{\Lambda}^T, \boldsymbol{\Psi}). \tag{2.5}$$

One of the biggest limitations of the factor analysis model is that it assumes the data is homogeneous or comes from a single distribution. In Section 2.1.4 we will describe a latent variable model that is more suited to heterogeneous data. The next section discusses a practical issue that must be taken into consideration when utilising the linear Gaussian factor model.

Factor Rotations

The linear Gaussian factor model as specified above is indeterminate and does not specify a unique set of parameters but a multiplicity of parameter sets, each related to the other by an orthogonal transformation [49]. The factor model decomposes the sample covariance matrix of the data, $\boldsymbol{\Sigma}$, as $\boldsymbol{\Lambda}\boldsymbol{\Lambda}^T + \boldsymbol{\Psi}$. However if we consider a non-singular orthogonal transformation of $\boldsymbol{z}$ to new latent variables $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{z}$, then the y_i will still be standard Gaussian and

$$\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{\Lambda}\boldsymbol{M}^T\boldsymbol{y} + \boldsymbol{\epsilon},$$

so that:

$$\boldsymbol{\Sigma} = \boldsymbol{\Lambda}\boldsymbol{M}\boldsymbol{M}^T\boldsymbol{\Lambda}^T + \boldsymbol{\Psi} = \boldsymbol{\Lambda}\boldsymbol{\Lambda}^T + \boldsymbol{\Psi}.$$

Thus our latent variables are indeterminate up to an orthogonal transformation [80]. Geometrically speaking the columns of $\boldsymbol{\Lambda}$ can be viewed as defining the axes of the lower dimensional latent space (coordinate system) of factors. Since a rotation is a non-singular orthogonal transformation and a permutation of columns is a particular type of rotation we say that the factor solution is invariant to permutation of axes. In situations where this rotation invariance becomes problematic, for example when trying to compute the average of a number of factor loading matrices, the Procrustes transformation will be utilised. A Procrustes transformation [28, 73] is a geometric transformation that involves only translation, rotation, uniform scaling or a combination of these transformations. Hence it may change the size but not the shape of a geometric object. This will require the definition of a

template orientation to which all other configurations will be mapped using the Procrustes transformation, allowing for coherent averaging.

2.1.2 Principal Component Analysis

Principal component analysis (PCA) [64, 104] has proven to be an exceedingly popular technique for dimensionality reduction that is discussed in most texts on multivariate analysis [14, 16] and in great detail in the exemplar reference [71]. Its many application areas include data compression [132], image analysis, visualisation [71], pattern recognition [75], regression and time series prediction.

The most common definition of PCA is that for a set of observed p-dimensional data vectors, $\boldsymbol{X}$, the q principal axes $\boldsymbol{w}_j, j \in \{1, \ldots, q\}$, are those orthonormal axes onto which the retained variance under linear projection is maximal [64]. The main attraction of PCA is its computational simplicity as all it requires is an eigen-decomposition of the sample covariance or correlation matrix to obtain the largest eigenvalues and eigenvectors [14]. A limiting disadvantage of PCA as defined above is the absence of an associated probability density or generative model, it is purely a rotation, which arbitrarily discards low value components. To overcome this Probabilistic PCA [131] was developed based on a latent variable model framework. To carry out PCA all that is required is the eigen-decomposition of the covariance matrix:

$$\boldsymbol{\Sigma} = \boldsymbol{A}\boldsymbol{W}\boldsymbol{A}^T,$$

where $\boldsymbol{W}$ is a diagonal matrix whose elements are the eigenvalues of $\boldsymbol{\Sigma}$ and $\boldsymbol{A}$ is an orthogonal matrix whose columns are the corresponding eigenvectors of $\boldsymbol{\Sigma}$. However in Section 2.1.1 the factor analysis decomposition of $\boldsymbol{\Sigma}$ is defined as:

$$\boldsymbol{\Sigma} = \boldsymbol{\Lambda}\boldsymbol{\Lambda}^T + \boldsymbol{\Psi},$$

which is equivalent to the PCA decomposition when

$$\boldsymbol{\Lambda} = \boldsymbol{A}\boldsymbol{W}^{\frac{1}{2}} \qquad \text{and} \qquad \boldsymbol{\Psi} = \text{diag}(0, \ldots, 0)_p.$$

From Equation 2.5 we can see that if, $\boldsymbol{\Psi} = \text{diag}(0, \ldots, 0)_p$, the conditional distribution of $\boldsymbol{x}$ is now degenerate with all the probability concentrated at the mean [9]. This gives us an insight into one of the main advantages of FA over PCA, in that FA can model covariance amongst input dimensions separately from variance, by inflating $\boldsymbol{\Psi}$, whereas PCA cannot. It also highlights the more statistically sound model based approach of FA which does not result in degenerate probabilities. In the case where the q eigenvalues of $\boldsymbol{\Sigma}$ are large and the remainder small, PCA can produce a decomposition that is similar to the solution obtained by factor analysis but in general they will give quite different results due to their inherently different properties [62].

In response to the shortcomings mentioned above a more principled model based approach to PCA was developed in [130]. This is known as Probabilistic Principal Components Analysis (PPCA) and is even more closely related to FA but makes the assumption of isotropic residuals, i.e.

$$\boldsymbol{\Psi} = \sigma^2 \boldsymbol{I},$$

which allows for a certain amount of noise due to the unused components but is constrained to be the same across each of the observed variables.

2.1.3 Canonical Correlation Analysis

Canonical Correlation Analysis (CCA) [65] is a standard tool from multivariate analysis which will feature heavily in Chapter 6. CCA is a method for measuring the linear relationship between two sets of multivariate observations [15]. In a sense it is similar to PCA where we are concerned with finding a linear transformation such that the components of the transformed vector are uncorrelated. Given two random vectors, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, of dimension m_1 and m_2, CCA is concerned with finding a *pair* of linear transformations such that one component within each set of transformed variables is correlated with a single component in the other set. The correlation matrix between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is reduced to a block diagonal matrix with blocks of size two where each block is of the form $\begin{pmatrix} 1 & \rho_i \\ \rho_i & 1 \end{pmatrix}$, potentially padded with the identity matrix if $m_1 \neq m_2$. The non-negative numbers ρ_i, at most $p = \min(m_1, m_2)$ of which are nonzero, are called the canonical correlations and are

usually ordered from largest to smallest [7].

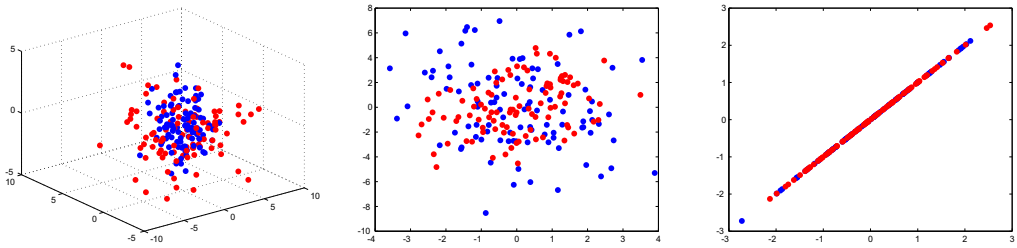
In order to illustrate the effects of CCA a toy example using synthetic data is presented, the results of which are shown in Figure 2.2. Two data sets, $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ were generated. The data set $\boldsymbol{X}_1$ is composed of 100 samples from a multivariate Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$:

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & .1 & .5 \\ .1 & 1 & .3 \\ .5 & .3 & 1 \end{pmatrix}.$$

The data set $\boldsymbol{X}_2$ is generated as a linear combination of the dimensions of $\boldsymbol{X}_1$:

$$\boldsymbol{X}_2 = \boldsymbol{X}_1 \times \begin{pmatrix} 1 & 1 & 1 \\ -2 & .9 & 0 \\ 1 & 0 & -1 \end{pmatrix}$$

The first and second principal components of $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ were extracted using PCA and are plotted in Figure 2.2b. Despite the fact $\boldsymbol{X}_2$ is linearly dependent on $\boldsymbol{X}_1$ no relationship is apparent from the principal components, this is because PCA seeks the transformation that maximises the variance *within* each dataset, $\boldsymbol{X_1}$ or $\boldsymbol{X_2}$, individually . In contrast the relationship between the canonical variates, the points obtained by transforming the original points by their respective weight matrices estimated by CCA, plotted in Figure 2.2c is clearly linear. This is because CCA seeks the transformations that maximise the covariance *between* both datasets $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$. As with PCA the CCA problem can be computed by solving a generalised eigenvalue problem [76] or alternatively it can be formulated as a latent variable model. In the latter model the two random vectors $\boldsymbol{x}_1$ and $\boldsymbol{x_2}$ are considered to have been generated by some unknown transformations, $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$, of the same latent variable $\boldsymbol{z}$ subject to independent Gaussian noise, $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_1$. The correlations between the two data sets are accounted for by the shared latent or source variable and the model takes



(a) Data (b) Principal components (c) Canonical variates

Figure 2.2: Canonical Correlation Analysis example see text for details. (a): data $\boldsymbol{X}_1$, in blue, and $\boldsymbol{X}_2$, in red, plotted in 3-D. (b): First two principal components of $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ plotted against each other with the blue points corresponding to the first principal component and the red the second. (c): the canonical variates obtained from CCA and reveals the linear dependence between the two data sets.

the following form [7]:

$$\boldsymbol{x_1} = \boldsymbol{\mu}_1 + \boldsymbol{z}W_1^T + \boldsymbol{\epsilon}_1, \tag{2.6}$$

$$\boldsymbol{x_2} = \boldsymbol{\mu}_2 + \boldsymbol{z}W_2^T + \boldsymbol{\epsilon}_2, \tag{2.7}$$

$$p(\boldsymbol{z}) \sim \mathcal{N}(\boldsymbol{z}|\mathbf{0}_p, \boldsymbol{I}_p), \tag{2.8}$$

$$p(\boldsymbol{\epsilon}_1) \sim \mathcal{N}(\boldsymbol{\epsilon}_1|\mathbf{0}_p, \boldsymbol{\Psi}_1), \tag{2.9}$$

$$p(\boldsymbol{\epsilon}_2) \sim \mathcal{N}(\boldsymbol{\epsilon}_2|\mathbf{0}_p, \boldsymbol{\Psi}_2). \tag{2.10}$$

The maximum likelihood estimates of the parameters of this model lead to the canonical correlation directions [137]. The model looks similar to the latent variable model formulation of FA and PCA but the core difference is in the full covariance matrices $\boldsymbol{\Psi}_1$ and $\boldsymbol{\Psi}_2$ introduced instead of the simple spherical noise model, $\boldsymbol{\Psi} = \sigma^2\mathbf{I}$, of PCA or diagonal noise model, $\boldsymbol{\Psi} = \text{diag}(\sigma^2)$, of factor analysis. This is necessary for CCA to be able to focus on modeling the correlations but at the same time it poses notable computational difficulties in real applications [7].

The probabilistic formulation of CCA is merely an alternative but equivalent formulation of the original or classical CCA, however it makes justified extensions possible [137] and in the next section we will discuss a Bayesian approach to estimating the model.

Bayesian CCA

The probabilistic model described by Equations 2.6-2.10 is here extended to a Bayesian generative model by introducing suitable prior distributions. We adopt the formulation of [137, 76] and the prior distributions for the model parameters are:

$$\boldsymbol{w}_i \sim \mathcal{N}(\mathbf{0}_p, \alpha_i \boldsymbol{I}), \tag{2.11}$$

$$\alpha_i \sim \text{Gam}(\alpha_0, \beta_0), \tag{2.12}$$

$$\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2 \sim \text{IW}(\boldsymbol{I}, v_0), \tag{2.13}$$

$$\boldsymbol{\mu_1}, \boldsymbol{\mu_2} \sim \mathcal{N}(\mathbf{0}_p, \sigma^2 \boldsymbol{I}), \tag{2.14}$$

where $\boldsymbol{w}_i$ denotes the ith column of $\boldsymbol{W}_1$ or $\boldsymbol{W}_2$. The conditional dependencies of the model are illustrated using a Directed Acyclic Graph (DAG) in Figure 2.3 The priors for the mean, $\boldsymbol{\mu}$, and the covariance matrices, $\boldsymbol{\Psi}$, are standard conjugate priors and the prior for the transformation matrices is the Automatic Relevance Determination (ARD) prior [14]. The purpose of the ARD prior is to automatically control the number of components extracted by the model. The parameter α_i controls the magnitude of $\boldsymbol{w}_i$, if the dimensionality of the dependent subspace is less than the full dimensionality of $\boldsymbol{W}$, then α_i for the remaining columns goes towards infinity, and the actual elements of the vectors go to zero. This will prove useful in Chapter 6 as a means of determining whether or not there is correlation between two data sets and this section will be referred to again as part of that work.

The model can be estimated using variational approximations [135, 137]. The variational approach is a deterministic approximation method that generally provides bounds on conditional or marginal probabilities and will be discussed in Section 2.3.2

2.1.4 Finite Mixture Models

Another well-known example of a latent variable model is the mixture distribution in which the latent variable is the discrete component label [130]. Finite mixture models [14, 82] provide a rich class of models that are heavily used in statistical modelling and that have

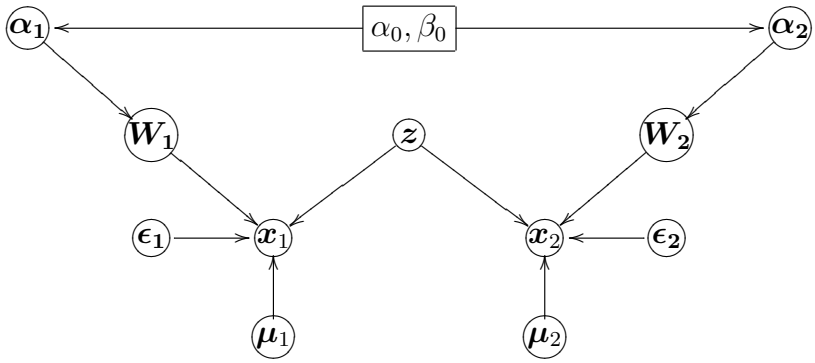


Figure 2.3: Directed Acyclic Graph of the Bayesian CCA model.

been extensively studied in recent years by both the Neural Computation and Machine Learning communities for a variety of applications. The use of finite mixture models is particularly relevant to applications where the input space is assumed to be heterogeneous, so that it would be unrealistic to use a single density to model the distribution of the data [49]. The description of a finite mixture of distributions is straightforward: any convex combination of distributions is a mixture,

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k f_k(\boldsymbol{x}; \boldsymbol{\theta}_k), \tag{2.15}$$

where the f_k are densities and the π_k are non-negative real numbers that sum to one; that is:

$$0 \leq \pi_k \leq 1, \qquad (k = 1, \ldots, K), \qquad \sum_{k=1}^{K} \pi_k = 1. \tag{2.16}$$

The quantities $\pi_1, \ldots, \pi_K$ are called the *mixing proportions* or weights. The $f_k(\boldsymbol{x})$ are called the *component densities* of the mixtures. Later when we use the term component, we mean a specific component and the corresponding parameters, $\boldsymbol{\theta}_k$. In most cases the f_k's are from the same parametric family, with unknown parameter $\boldsymbol{\theta}_k$, leading to the parametric

mixture model:

$$\sum_{k=1}^{K} \pi_k f(\boldsymbol{x}|\boldsymbol{\theta}_k). \tag{2.17}$$

Mixture models are commonly used for density estimation as they combine the flexibility of non-parametric methods while retaining some of the advantages of parametric approaches, they are thus described as being *semi-parametric* [82]. The latent variable framework described earlier is a key feature of the model. Defining a joint distribution over observed and latent variables, the corresponding distribution of the manifest variables alone is obtained by marginalisation. This allows relatively complex marginal distributions over manifest variables to be expressed in terms of more tractable joint distributions over the expanded space of manifest and latent variables. The introduction of latent variables thereby allows complicated distributions to be formed from simpler components [14], in the next section the latent variable or missing data formulation of the finite mixture model is discussed.

Missing Data Formulation

It is convenient when working with finite mixture models to introduce the missing data formulation of the model. Each observation $\boldsymbol{x}$ has an associated K-dimensional binary latent variable, $\boldsymbol{z}$, in which a particular element z_k is equal to one and all other elements are equal to zero [82]. The discrete latent variable, $\boldsymbol{z}$, can be interpreted as defining assignments of data points to specific components of the mixture [14]. The marginal distribution over $\boldsymbol{z}$ is specified in terms of the mixing proportions such that

$$p(z_k = 1) = \pi_k, \tag{2.18}$$

and the $p(\boldsymbol{z})$ follows a multinomial distribution [49]. This allows us to work with the joint distribution of $\boldsymbol{x}$ and $\boldsymbol{z}$, instead of the marginal distribution of $\boldsymbol{x}$ which simplifies the mixture problem greatly. The conditional distribution of $\boldsymbol{x}$ given a particular value for $\boldsymbol{z}$ is:

$$p(\boldsymbol{x}|z_k = 1) = f(x|\theta_k). \tag{2.19}$$

The joint distribution is given by $p(\boldsymbol{z})p(\boldsymbol{x}|\boldsymbol{z})$ which are defined in Equations 2.18 and 2.19 and the marginal distribution of $\boldsymbol{x}$ is then obtained by summing the joint distribution over all possible states of $\boldsymbol{z}$ to give:

$$\begin{aligned} p(\boldsymbol{x}) &= \int p(\boldsymbol{z})p(\boldsymbol{x}|\boldsymbol{z})\mathrm{d}\boldsymbol{z} \\ &= \sum_{k=1}^{K} \pi_k f(\boldsymbol{x}|\theta_k) \end{aligned}$$

This is the same as the original formulation in Equation 2.17 however the incorporation of the latent variables significantly simplidies model estimation.

In order to make inference on component assignments we consider the conditional probability of $\boldsymbol{z}$ given $\boldsymbol{x}$ which, using Bayes' theorem, is given by [14]:

$$\begin{aligned} p(z_k = 1|\boldsymbol{x}, \pi_k, \boldsymbol{\theta}_k) &= \frac{p(z_k = 1)p(\boldsymbol{x}|z_k = 1)}{\sum_{k=1}^{K} p(z_k = 1)p(\boldsymbol{x}|z_k = 1)} \\ &= \frac{\pi_k f(\boldsymbol{x}|\boldsymbol{\theta}_k)}{\sum_{k=1}^{K} \pi_k f(\boldsymbol{x}|\boldsymbol{\theta}_k)}, \end{aligned}$$

where f is the specified component distribution, e.g. Gaussian, Student-t. We shall view π_k as the prior probability of $z_k = 1$ and the corresponding conditional posterior as the probability once the data has been observed. This conditional posterior can be considered the 'responsibility' that a component has for generating $\boldsymbol{x}$ and gives us a natural framework for carrying out clustering of the data, by computing these responsibilities [82]. These can be used in a 'hard' clustering framework where a point is assigned exclusively to the component with the highest responsibility or in a 'soft' clustering method where we allow for joint membership weighted by the responsibility.

The mixture model with Gaussian densities can be combined with the linear Gaussian factor model described previously in Section 2.1.1 to create a Mixtures of Factor Analysers model which will be discussed in detail in Section 3.1.

2.2 Model Selection

Fitting the mixture model introduced in the previous section requires the specification of K, the number of components in the mixture. Ideally K is to be inferred from the data as part of a fully unsupervised approach, rather than specified a priori as in a K-means clustering method. Inference over K is effectively the problem of choosing from a set of competing models, e.g. the mixture model with $K = 4$ over the model with $K = 5$. This model selection problem is an issue of key interest in the clustering application presented in Section 3.5 where it provides insight to the number of unknown sub-populations within a heterogeneous sample.

Standard model estimation techniques such as a maximum likelihood approach to parameter estimation, i.e. choose the value of K that gives the highest value for the likelihood function, are inappropriate in this setting. This is because adding components to the model or increasing the model complexity, always increases the likelihood [82]. As a result alternative methods must be employed.

Generally we are interested in using the model to make predictions about future events or unseen data and thus the predictive power of the model is of interest. As an example having trained a model based on the sample data we may wish to use this model to predict the component assignment, $p(\boldsymbol{z})$, for a new point, $\boldsymbol{x}_{n+1}$

$$p(z_{n+1,k} = 1|\boldsymbol{x}_{n+1}, \boldsymbol{\theta}) = \frac{p(z_{n+1,k} = 1|\boldsymbol{\theta})p(\boldsymbol{x}_{n+1}|\boldsymbol{\theta}, z_{n+1,k} = 1)}{\sum_{k=1}^{K} p(z_{n+1,k} = 1|\boldsymbol{\theta})p(\boldsymbol{x}_{n+1}|\boldsymbol{\theta}, z_{n+1,k} = 1)}.$$

Increasing the model complexity however degrades its predictive performance as the model becomes too finely tuned to the training data. This is known as overfitting [14]. A number of measures have been devised to tackle this overfitting problem. An additional term can be introduced to the likelihood that penalises model complexity. There are many different measures available such as the Akaike Information Criterion [2] or Efron Information Criterion [36]. The disadvantage of such an approach is that the choice of measure is quite arbitrary and their criteria and sensitivity varies.

Another approach is to employ computer intensive data re-use techniques such as cross-validation [77, 106]. Here some of the data is held back from the training data and the

model that best predicts the unseen data is used.

These methods both have their shortcomings, different choices of penalisation terms can produce different results and cross-validation techniques greatly increase the amount of computation required. Also in situations where data is limited it is undesirable to have to throw away a portion of it. In the next section we will show how taking a Bayesian approach to model selection resolves these issues and provides a natural penalisation for excessively complex models.

2.2.1 Bayesian Model Selection

The Bayesian view of model comparison simply involves the use of probabilities to represent uncertainty in the choice of model, along with consistent application of the sum and product rules of probability [83]. Our uncertainty in a choice of models, $\{\mathcal{M}_i\}$ where $i = 1, 2, 3 \ldots$, is expressed through a prior distribution $p(\mathcal{M}_i)$. Given a data set $\mathcal{D}$ we then wish to evaluate the posterior distribution

$$p(\mathcal{M}_i|\mathcal{D}) \propto p(\mathcal{M}_i)p(\mathcal{D}|\mathcal{M}_i). \tag{2.20}$$

The prior allows us to express a preference for different models. Let us simply assume that all models are given equal prior probability. The interesting term in Equation 2.20 is the *model evidence*, $p(\mathcal{D}|\mathcal{M}_i)$, which expresses the preference shown by the data for different models. The model evidence is also referred to as the marginal likelihood because it can be viewed as a likelihood function over the space of models, in which the parameters have been marginalised out [14]. The ratio of model evidences, $p(\mathcal{D}|\mathcal{M}_i)/p(\mathcal{D}|\mathcal{M}_j)$, for two models is known as a Bayes factor [72] and can be used to compare pairs of models.

For a model governed by a set of parameters $\boldsymbol{\theta}$ the model evidence is given by

$$p(\mathcal{D}|\mathcal{M}_i) = \int p(\mathcal{D}|\boldsymbol{\theta}, \mathcal{M}_i)p(\boldsymbol{\theta}|\mathcal{M}_i)d\boldsymbol{\theta}.$$

This marginalisation over parameters neatly encapsulates the principle of Occam's razor, that the simplest possible model that explains the data should be used [83]. Figure 2.4 gives the basic intuition for why complex models can turn out to be less probable.

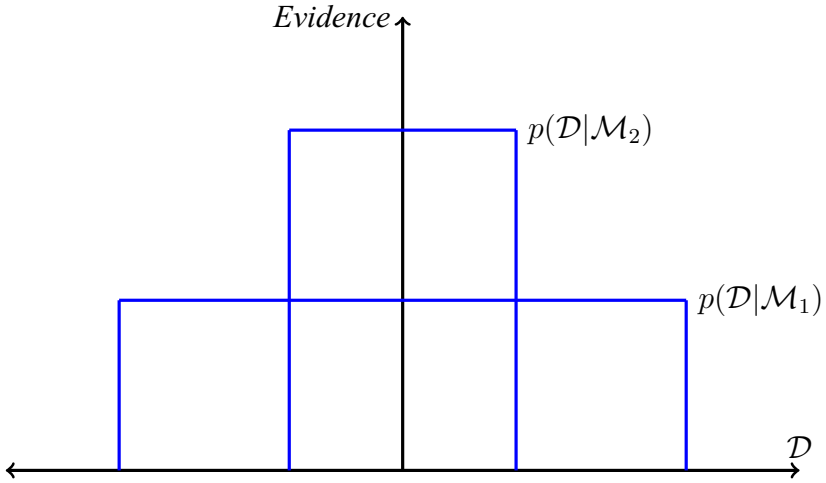


Figure 2.4: The horizontal axis represents the space of possible data sets, $\mathcal{D}$. The probability of the data given the model, $\mathcal{M}_i$, $p(\mathcal{D}|\mathcal{M}_i)$, is called the evidence for $\mathcal{M}_i$.

The horizontal axis is a one dimensional representation of the space of possible data sets, so that each point on this axis corresponds to a specific data set. Bayes' theorem rewards models in proportion to how well they predicted the data that occurred [14]. These predictions are quantified by the evidence, the probability of the data given the model. The simpler model, $\mathcal{M}_2$, makes only a limited range of predictions as shown by $p(\mathcal{D}|\mathcal{M}_2)$. The more powerful model, $\mathcal{M}_1$, is able to predict a greater variety of data sets, as seen by the greater width of the x-axis covered by $\mathcal{M}_1$. This means, however, that $\mathcal{M}_1$ does not predict the data sets in regions close to the vertical axis as strongly as $\mathcal{M}_2$. If equal prior probabilities have been assigned to both models and the data falls within this region, the less flexible model, $\mathcal{M}_2$, will be the more probable model [83].

The model evidence makes it possible to choose between two competing models while taking model complexity into account. In Chapter 3 a two tiered model selection problem will be encountered with both K, the number of components, and q_k, the number of factors in each individual component to be inferred from the data. The different combinations of K and q_k gives us a huge number of models to compare. Clearly, with the explosion in the number of possible models, computing the model evidence and comparing Bayes factors for each model is infeasible. Instead stochastic methods will be employed to compute a search over the model space which finds more likely configurations. This will be discussed further in Section 3.2.

2.3 Estimation Methods

For the latent variable models described earlier in this chapter we will be required to evaluate the posterior distribution, $p(\boldsymbol{Z}|\boldsymbol{X})$, of the latent variables given the observed data, $\boldsymbol{Z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)^T$ and $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ respectively. In this thesis a fully Bayesian approach will be employed to inference in all models. This makes model estimation a difficult procedure and for many of the models used it will be infeasible to evaluate the posterior distribution or indeed to even compute expectations with respect to this distribution. This may either be because the posterior distribution has a highly complex form for which expectations are not analytically tractable or that the required integrations may not have closed form analytical solutions.

Approximation schemes are required to overcome these model estimation challenges. Those used in this thesis fall into two classes, stochastic approximation techniques such as Markov Chain Monte Carlo [54, 91] and deterministic approximation schemes such as variational Bayes [6, 14]. These are extensive topics with a vast catalogue of literature behind them and a detailed exploration of the theory of both is beyond the scope of this thesis but can be found in the texts referenced above. Here we seek to provide a general overview and key definitions that will be required to understand the specific instances utilised in later chapters. We begin with MCMC.

2.3.1 Markov Chain Monte Carlo

Stochastic techniques such as Markov Chain Monte Carlo (MCMC) have enabled the widespread use of Bayesian methods across many domains [54]. They generally have the property that given infinite computational resource they can generate exact results and the approximation arises from the use of a finite amount of processor time [14]. MCMC is a strategy for generating samples $\boldsymbol{x}^i$ while exploring the state space $\boldsymbol{\chi}$, the space of all possible configurations or states, using a Markov chain mechanism [5]. It consists of Monte Carlo integration using Markov chains and each element will be introduced individually, starting with Monte Carlo integration.

Monte Carlo integration evaluates the required integral, $\mathbb{E}[f(\boldsymbol{x})]$, by drawing samples,

$\{\boldsymbol{x}^i, i = 1, \ldots, n\}$, from the posterior distribution and then approximating:

$$\mathbb{E}[f(\boldsymbol{x})] \approx \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}^i). \tag{2.21}$$

So the population mean of $f(\boldsymbol{x})$ is estimated by a sample mean. When the samples $\boldsymbol{x}^i$ are independent the laws of large numbers ensure that the approximation can be made as accurate as desired by increasing, n, the sample size [54]. In general drawing samples $\boldsymbol{x}^i$ independently from the posterior is not feasible since it can be quite non-standard. However the samples need not necessarily be independent. The $\boldsymbol{x}^i$ can be generated by any process which draws samples throughout the support of the posterior distribution in the correct proportion. This can be achieved using a Markov chain which has the target posterior as its stationary distribution and the combination of these methods is then Markov chain Monte Carlo.

It is intuitive to introduce Markov chains on finite state spaces, where $\boldsymbol{x}^i$ can only take s discrete values $\boldsymbol{x}^i \in \boldsymbol{\chi} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_s\}$. The stochastic process $\boldsymbol{x}^i$ is called a Markov chain if:

$$p(\boldsymbol{x}^i | \boldsymbol{x}^{i-1}, \ldots, \boldsymbol{x}^1) = T(\boldsymbol{x}^i | \boldsymbol{x}^{i-1}),$$

where T is a fixed transition matrix composed of the probabilities associated with various state-changes [113]. The chain is *homogeneous* if T remains invariant for all i, with $\sum_{x^i} T(\boldsymbol{x}^i | \boldsymbol{x}^{i-1} = 1)$ for any i. For any starting point the chain will converge to the invariant distribution $p(\boldsymbol{x})$, as long as T is a stochastic transition matrix that obeys the following properties [5]:

1. *Irreducibility*. For any state of the Markov chain there is a positive probability of visiting all other states. That is, T cannot be reduced to separate smaller matrices.

2. *Aperiodicity*. The chain should not get trapped in cycles.

A sufficient (but not necessary) condition to ensure that a particular $p(\boldsymbol{x})$ is the desired invariant distribution is to choose the transition probabilities to satisfy the property of *de-*

tailed balance, defined by:

$$p(\boldsymbol{x}^i)T(\boldsymbol{x}^{i-1}|\boldsymbol{x}^i) = p(\boldsymbol{x}^{i-1})T(\boldsymbol{x}^i|\boldsymbol{x}^{i-1}).$$

A Markov chain that respects detailed balance is said to be reversible [14].

MCMC samplers are irreducible and aperiodic Markov chains that have the target distribution as the invariant distribution [5]. Two specific examples, the Metropolis-Hastings and Gibbs samplers are introduced in the next sections. It should be emphasised that the samples obtained from Markov chains are not a set of independent samples from the invariant distribution. This is because successive samples are highly correlated [113]. To obtain independent samples it is necessary to discard most of the sequence and only retain every M^{th} sample, a process called 'thinning'. Another issue for practical consideration is the dependence on the initial value of the chain $\boldsymbol{x}_0$. In order to remove this dependence a burn-in period of sufficient length is required at the start of sampling and samples from within the burn-in period are discarded [54].

Metropolis Hastings

The Metropolis-Hastings algorithm [90, 59] is a method used to construct a Markov chain such that its stationary distribution is precisely our distribution of interest. For the Metropolis-Hastings algorithm at each sample i, the next state $\boldsymbol{x}^{i+1}$ is chosen by first sampling a candidate point $\boldsymbol{x}^*$ from a proposal distribution $q(.|\boldsymbol{x}^i)$. The candidate point is then accepted with probability $\alpha(\boldsymbol{x}^i, \boldsymbol{x}^*)$ where:

$$\alpha(\boldsymbol{x}^i, \boldsymbol{x}^*) = \min\left(1, \frac{p(\boldsymbol{x}^*)q(\boldsymbol{x}^i|\boldsymbol{x}^*)}{p(\boldsymbol{x}^i)q(\boldsymbol{x}^*|\boldsymbol{x}^i)}\right).$$

If the candidate point is accepted, the next state becomes $\boldsymbol{x}^{i+1} = \boldsymbol{x}^*$. If the candidate point is rejected the chain does not move, $\boldsymbol{x}^{i+1} = \boldsymbol{x}^i$. The specific choice of proposal distribution can have a marked effect on the performance of the algorithm [14]. For continuous state spaces a common choice is a Gaussian centred on the current state, leading to an important trade off in determining the variance parameter of this distribution. If the variance is small then the proportion of accepted transitions will be high, but progress through the state

space takes the form of a slow random walk leading to long correlation times. However if the variance parameter is too large then the rejection rate will be high because in the kind of complex problems we are considering , many of the proposed steps will be to states for which the probability is low. In practice this parameter is adjusted to try and achieve an acceptance rate of approximately 23% which has been shown to be optimal for an n-dimensional Gaussian target distribution [111].

The Gibbs Sampler

Gibbs sampling [51] is a simple and widely applicable MCMC algorithm that can be seen as a special case of the Metropolis-Hastings algorithm where the proposals are always accepted. Thus rather than evaluating an acceptance probability, Gibbs sampling simply consists of draws from the posterior conditional of the variables of the target distribution drawn sequentially, each conditioned upon the previous. There will be some cases where it is not possible to determine the conditional distributions of the variables and in these cases it will not be possible to Gibbs sample, rather we will have to use the full Metropolis-Hastings algorithm as in Chapter 4.

2.3.2 Variational Bayesian Approximation

Variational methods have their origins in the 18th century with the work of Euler, Lagrange and others on the calculus of variations [14]. In order to describe these methods we first need to introduce the concept of a functional which is a mapping that takes a function as the input and that returns the value of the functional as the output, an example being entropy which takes a probability distribution $p(x)$ as an input and returns the quantity:

$$\mathrm{H}(p) = \int p(x) \ln p(x) \mathrm{d}x.$$

The derivative of the functional then expresses how the value of the functional changes in response to infinitesimal changes to the input function. Many problems can be expressed in terms of an optimisation problem in which the quantity being optimised is a functional. Although there is nothing intrinsically approximate about variational methods they do natu-

rally lend themselves to finding approximate solutions. This is done by restricting the range of functions over which the optimisation is performed. In the applications to probabilistic inference the restriction may for example take the form of factorisation assumptions. This factorised form of variational inference corresponds to an approximation framework developed in physics called mean field theory [101].

We now describe how variational optimisation can be applied to the inference problem for Bayesian models, such as those described earlier, consisting of both latent and manifest variables, $\boldsymbol{Z}$ and $\boldsymbol{X}$, and parameters $\boldsymbol{\theta}$. Good introductions to the subject of variational Bayesian approximations can be found in [11, 13, 133].

The probabilistic model specifies the joint distribution $p(\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{\theta})$ and our goal is to find an approximation for the posterior distribution $p(\boldsymbol{Z}, \boldsymbol{\theta}|\boldsymbol{X})$ as well as for the model evidence $p(\boldsymbol{X})$. The log marginal likelihood of the data can be lower bounded by introducing any distribution over both latent variables and parameters and appealing to Jensen's inequality (due to the concavity of the logarithm function):

$$\begin{aligned}\ln p(\boldsymbol{X}) = \ln \int p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta}) \mathrm{d}\boldsymbol{Z} \mathrm{d}\boldsymbol{\theta} &= \ln \int q(\boldsymbol{Z}, \boldsymbol{\theta}) \frac{p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta})}{q(\boldsymbol{Z}, \boldsymbol{\theta})} \mathrm{d}\boldsymbol{Z} \mathrm{d}\boldsymbol{\theta}, \\ &\geq \int q(\boldsymbol{Z}, \boldsymbol{\theta}) \ln \frac{p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta})}{q(\boldsymbol{Z}, \boldsymbol{\theta})} \mathrm{d}\boldsymbol{Z} \mathrm{d}\boldsymbol{\theta}.\end{aligned}$$

This does not simplify the problem since evaluating the true posterior distribution requires knowing its normalising constant [11]. Instead we use a simpler factorised approximation to $q(\boldsymbol{Z}, \boldsymbol{\theta}) \approx q_z(\boldsymbol{Z}), q_\theta(\boldsymbol{\theta})$.

$$\ln p(\boldsymbol{X}) \geq \int q_z(\boldsymbol{Z}), q_\theta(\boldsymbol{\theta}) \ln \frac{p(\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\theta})}{q_z(\boldsymbol{Z}), q_\theta(\boldsymbol{\theta})} \mathrm{d}\boldsymbol{Z} \mathrm{d}\boldsymbol{\theta} = \mathcal{F}(q_z(\boldsymbol{Z}), q_\theta(\boldsymbol{\theta}), \boldsymbol{X}) \tag{2.22}$$

The quantity $\mathcal{F}$ is a functional of the free distributions $q_z(\boldsymbol{Z})$ and $q_\theta(\boldsymbol{\theta})$. The variational Bayesian algorithm iteratively maximises $\mathcal{F}$ in Equation 2.22 with respect to the free distributions $q_z(\boldsymbol{Z})$ and $q_\theta(\boldsymbol{\theta})$.

Update equations can be derived for the latent variables and model parameters by taking expectations of the free distributions over all variables with respect to the other factors. If for the time being we incorporate the model parameters, which are treated as random variables anyway, into $\boldsymbol{Z}$ so that the elements of $\boldsymbol{Z}$ are partitioned into disjoint groups that

are denoted by $\boldsymbol{Z}_i$ our previous factorisation assumption means

$$q(\boldsymbol{Z}) = \prod_{i=1}^{M} q_i(\boldsymbol{Z}_i), \tag{2.23}$$

a general expression for the optimal solution $q_j^*(\boldsymbol{Z}_j)$ is given by [14]:

$$\ln q(\boldsymbol{Z}_j) = \mathbb{E}_{i \neq j}[\ln p(\boldsymbol{X}, \boldsymbol{Z})] + \text{const.} \tag{2.24}$$

This solution provides the basis for applications of variational methods. It says that the log of the optimal solution for factor $q_j(\boldsymbol{z}_j)$ is obtained simply by considering the log of the joint distribution over all hidden and observed variables and then taking the expectation with respect to all of the other factors $q_i(\boldsymbol{Z}_i)$ for all $i \neq j$. The set of equations given by Equation 2.24 for all $j = 1, \ldots, m$ represent a set of consistency conditions for the maximum of the lower bound subject to the factorisation of 2.23 [14]. They do not provide an explicit solution since they depend on the other factors $q_i(\boldsymbol{Z}_i)$ for $i \neq j$. Therefore a consistent solution is found by cycling through these factors and replacing each in turn with the revised estimate. Convergence of $\mathcal{F}$ is guaranteed because the bound is convex with respect to each of the factors $q_i(\boldsymbol{Z}_i)$ [133].

In Chapter 5 the variational method will be used to perform Bayesian inference over a latent variable model known as the Latent Position Cluster Model.

2.4 Conclusion

In this chapter, we have introduced some standard models for multivariate data analysis and discussed their properties from a latent variable model perspective. To add rigour to our model based approach we also described two methods for performing Bayesian model selection, providing us with a principled method for choosing the most appropriate model given the data. In the next chapter we carry out an empirical comparison of the Variational and MCMC model selection techniques and assess the conditions under which different performance is obtained between the two approaches. Models based on the latent variable framework discussed will feature throughout this thesis and in Chapter 3 we will utilise a

mixture of factor analysers to extract and analyse the different networks of dependencies in a heterogeneous population of autistic subjects.

Chapter 3

Simultaneous Clustering and Latent Structure Analysis using Mixtures of Factor Analysers

The previous chapter gave a general introduction to the concept of a latent variable model and discussed some specific examples. This chapter focuses on a well known model from the literature that is a combination of two of these examples, the linear factor model and finite mixture models. This Mixtures of Factor Analysers (MFA) [53] overcomes some of the weaknesses associated with each model. The flexibility of the linear factor model is increased by incorporation of elements of mixture models allowing factor analysis to be performed on data that is non-linear or heterogeneous. Complementary to this effect, incorporation of the FA decomposition of the covariance matrix, outlined in Section 2.1.1, naturally produces model based clustering of the data using fewer parameters than a full mixture of Gaussians model while simultaneously performing local dimensionality reduction. Due to the difficulties associated with mixture models outlined in Chapter 2 we favour a Bayesian approach to model estimation and describe how this can be applied to the MFA model in Section 3.1.1. The use of a mixture model framework raises the question of how many components, K, are appropriate given the data, coupled with this now in the MFA is the choice of the number of factors. In our approach the number of factors in each component, q_k, is allowed to vary, rather than constrain all components to have the same number

of factors, $q_k = q$. This two tiered model selection problem greatly increases the number of possible models making standard model selection techniques such as Bayes Factors infeasible. Instead a stochastic approach is employed that searches over the space of all models.

Although some of the content of this chapter may be considered background material, as it pertains to models and methods that are well known in the statistics community, we prefer to include it here as specific to the chapter rather than the more general material outlined in Chapter 2. The chapter is organised as follows.

Section 3.1 introduces the MFA model and a full Bayesian treatment is described in Section 3.1.1. In Section 3.2 we describe in detail the stochastic method used to estimate the model inferring both K and q_k from the data, Birth-Death MCMC (BDMCMC). This method will also feature in Chapter 4 in a different setting but again applied to a latent variable model. The use of MCMC methods in a mixture model setting requires a careful treatment of an issue known as label switching and Section 3.2.4, describes the method employed here to resolve this problem. The full algorithm, MFA estimated by BDMCMC, is then deployed to conduct an empirical comparison of the stochastic estimation method with an alternative method for performing model selection and dimension reduction based on variational Bayesian approximation. The variational method for MFA was developed by [10] and is described Section 3.3 which builds on the overview of variational methods provided in Section 2.3.2.

The chapter concludes with the application of the MFA to a real world data set in Section 3.5. The data in question is a set of Autism Diagnostic Interview - Revised (ADI-R) [81] scores for a cohort of 625 subjects gathered as part of research conducted by the International Molecular Genetic Study of Autism Consortium (IMGSAC). The ADI-R is a diagnostic tool used to determine whether a subjects suffers from autism and is described in Section 3.5.2. Autism is a pervasive development disorder which exhibits a wide range of symptoms of varying severity referred to as the autism spectrum. Previous studies have reported difficulties due to the heterogeneous nature of the population which confounds genetic analysis in the search for autism phenotypes. This is the first application of the MFA to ADI-R data and the combination of clustering and latent structure analysis, in an unsupervised manner, may provide insight as to what characterises these different sub-

populations.

3.1 Mixtures of Factor Analysers

Chapter 2 introduced the latent variable models; Factor Analysis and finite mixture models. The main limitation of FA is that it is a linear model. Combining FA with the mixture model framework it is possible to obtain a global extension of the basic model, that simultaneously performs clustering and local dimensionality reduction. This MFA is a more flexible extension of the basic factor model capable of handling heterogeneous data [49]. Each component of the mixture model is a linear factor model and if we keep the same assumptions as used in both mixture models and the FA model, as described in Sections 2.1.1 and 2.1.4, the marginal density of $\boldsymbol{x}$ is given by:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k \boldsymbol{\Lambda}_k^T + \boldsymbol{\Psi}), \tag{3.1}$$

which is simply a finite mixture of K Gaussians where, as previously, the π_k are mixing proportions. By exploiting the FA parameterisation of covariance matrices (see Section 2.1.1) a mixture of factor analysers can be used to fit a mixture of Gaussians to correlated high dimensional data without requiring $O(p^2)$ parameters, or undesirable compromises such as axis aligned covariance matrices. The MFA model has recently been used for clustering of high dimensional micro-array data [89, 141] and has proved less susceptible to overfitting than the full mixture of Gaussians. The dimension of the latent space, q, or columns of $\boldsymbol{\Lambda}$, allow us to control the number of parameters required to fit the model. The model can be made more flexible by allowing the value of q to vary between mixture components, q_k, this allows each cluster to potentially occupy a different subspace of the high dimensional feature space. Figure 3.1 provides a simple example of data with this kind of feature. The data shown is composed of three different clusters in three dimensions. Each cluster was generated from a factor model with a different number of factors. The effect of this can be observed in the structure that they exhibit in the plot with the red single factor cluster having a cigar shape, the blue two factor cluster a sheet shape and the green three factor cluster a cloud shape. Clearly the use of a fixed factor to describe all clusters would

result in the loss of information about the structure of the clusters.

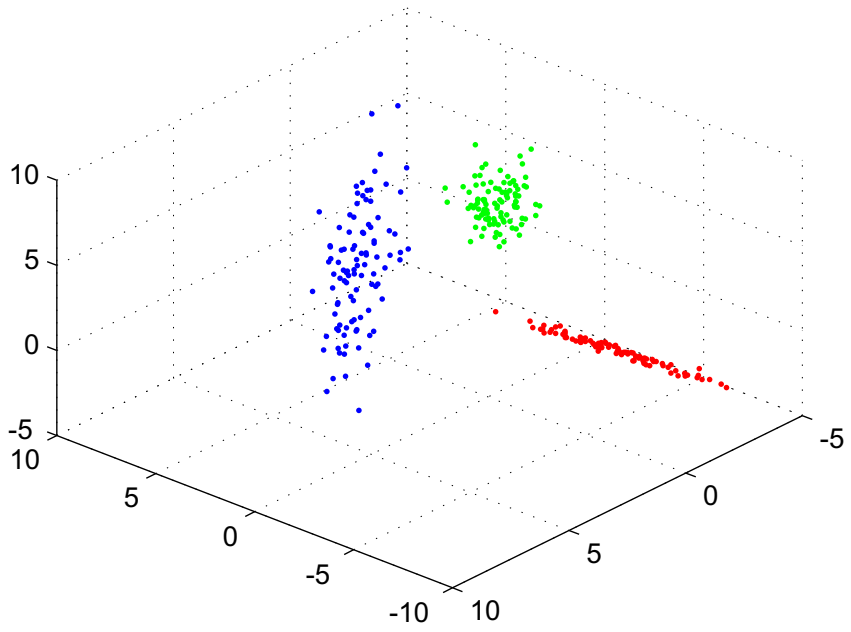


Figure 3.1: Three clusters inhabiting subspaces of different dimensions. The cigar shaped red cluster originated from a single factor factor model. The blue sheet shaped cluster was generated from a two factor model. The green cloud shaped cluster was generated from a three factor model.

This extension gives us a very powerful model-based, variable dimensional clustering algorithm that shares many characteristics with subspace clustering and feature selection algorithms for clustering. These algorithms simultaneously look for clusters within the data as well as the variables of the data that give the best clustering so as to remove redundant features that confuse analysis [102, 103]. The MFA model however is more of a feature transformation algorithm that finds clusters by capturing the dependency structure in the data through $\mathbf{\Lambda}_k$. Correct estimation of q_k is a key part of the clustering problem as with too few factors $\mathbf{\Lambda}_k$ is unable to capture enough of the dependence structure to uncover the differences between clusters [48]. In the other extreme case, with too many factors the MFA is able to model spurious correlations in the data and as such many clusters become one supercluster.

In the MFA formulation the independent disturbance vector, $\mathbf{\Psi}$, is assumed to be the

same for all components, when this is isotropic the model becomes a mixtures of PPCA [82].

3.1.1 Full Bayesian MFA

There are a number of pitfalls to consider when estimating the MFA. As with all mixture models the MFA suffers from a number of complications in the likelihood: it is susceptible to singularities resulting from components collapsing onto one data point [14]. Additionally, the likelihood is always increased by adding more components or factors so a maximum likelihood approach is not applicable without some form of penalisation (as discussed in Section 2.2). A Bayesian approach overcomes these problems by treating the parameters of the model as unknown random quantities and averaging over the ensemble of models they provide. This requires the specification of priors for the parameters of the model and we adopt the same structure used in [49]. A Direct Acyclic Graph is shown in Figure 3.2 showing the dependencies in the model. As the model is a combination of Factor Analysis and a mixture model there are two latent variables to consider, the factor scores (see Section 2.1.1) which will be referred to as $\boldsymbol{y}$ and the cluster allocation variable (see Section 2.1.4) denoted by $\boldsymbol{z}$.

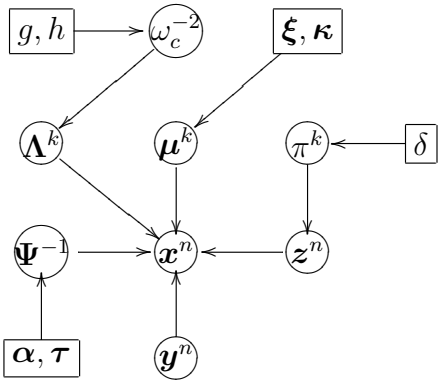


Figure 3.2: Direct Acyclic Graph of Bayesian MFA Model, squares denote fixed variables, circles denote random variables, k is used to index over components, n indexes over data points.

Conjugate priors are used for the parameters of the model and the distributions of all parameters are of the following form [49]:

$$
\begin{aligned}
p(\boldsymbol{\mu}_k) &\sim \mathcal{N}(\boldsymbol{\xi}, \boldsymbol{\kappa}), \\
p(\boldsymbol{\Lambda}_{kr}) &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Omega}), \\
p(\omega_c^{-2}) &\sim \text{Ga}(g, h), \\
p(\sigma_p^{-2}) &\sim \text{Ga}(\alpha, \tau), \\
p(\boldsymbol{\pi}) &\sim \text{Dir}(\boldsymbol{\delta}).
\end{aligned}
$$

A hierarchical prior is used for the loading matrices, $\boldsymbol{\Lambda}_k$, this adds extra flexibility to the model allowing the loading matrices for each cluster to be similar but not constrained to be the same. We define the column vector $\boldsymbol{\Lambda}_{kr}$ made up of the r-th row of the k-th matrix of factor loadings and use the zero mean Gaussian prior:

$$\boldsymbol{\Lambda}_{kr} \sim \mathcal{N}(\mathbf{0}_p, \boldsymbol{\Omega}),$$

for $k = 1, \ldots, K$ and $r = 1, \ldots, p$. The hyperparameter, $\boldsymbol{\Omega}$, is assumed to be diagonal and more precisely: $\boldsymbol{\Omega}^{-1}$ =diag$(\omega_1^{-2}, \ldots, \omega_q^{-2})$ with:

$$p(\omega_c^{-2}) \sim Ga(g, h), \qquad c = 1, \ldots, q.$$

The same $\boldsymbol{\Omega}$ is used for all components. Gamma distributions are used for the diagonal elements of the noise precision, $\boldsymbol{\Psi}^{-1}$ =diag$(\sigma_1^{-2}, \ldots, \sigma_p^{-2})$. A Dirichlet prior, which is the conjugate of the multinomial posterior distribution, is used for the mixing proportions where $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_K)$. Initially we assume that it is equally likely for the data to have arisen from each cluster and therefore set $\delta_1 = \delta_2 =, \ldots, \delta_K = \delta$.

Using the specified conjugate priors it is possible to explicitly calculate full conditional posterior distributions for the model under the assumption of fixed K and $q_k = q$. A two stage Gibbs sampler, sometimes referred to as a data augmentation algorithm, can then be used to draw samples from the conditional posterior distributions of these parameters. The first stage of this algorithm imputes values for the missing or unobserved parameters, $\boldsymbol{z}_i, \boldsymbol{y}_i$, while the second stage performs the estimation on the complete data. This is analogous to

the standard deterministic EM algorithm. The first step is to simulate samples from the conditional posterior distributions of the latent variables, which are the latent allocation variables:

$$\boldsymbol{z}_i \sim \text{Mn}(1, \pi^*_{1i}, \ldots, \pi^*_{Ki}) \text{ with } \pi_{ki} \sim \pi_k \mathcal{N}(\boldsymbol{x}_i; \boldsymbol{\mu}_k + \boldsymbol{y}_i \boldsymbol{\Lambda}_k^T, \boldsymbol{\Psi}),$$

and the factor scores:

$$\boldsymbol{y}_{i:z_i=k} \sim \mathcal{N}\big((\mathbf{I} + \boldsymbol{\Lambda}_k^T \boldsymbol{\Psi}^{-1} \boldsymbol{\Lambda}_k)^{-1} \boldsymbol{\Lambda}_k^T \boldsymbol{\Psi}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T, (\mathbf{I} + \boldsymbol{\Lambda}_k^T \boldsymbol{\Psi}^{-1} \boldsymbol{\Lambda}_k)^{-1}\big).$$

In the second stage the observable parameters are estimated, starting with the mixing proportions:

$$\boldsymbol{\pi} \sim \text{Dir}(\delta + n_1, \ldots, \delta + n_K),$$

where $n_k = \#\{i : z_i = k\}$ for $k = 1, \ldots, K$ denotes the number of observations currently allocated to component k of the mixture. For each $\boldsymbol{\mu}_k$ a Gaussian prior is used which leads to a Gaussian conditional posterior:

$$\boldsymbol{\mu}_k \sim \mathcal{N}((n_k \boldsymbol{\Psi}^{-1} + \boldsymbol{\kappa}^{-1})^{-1} (n_k \boldsymbol{\Psi}^{-1} \hat{\boldsymbol{x}}_k + \boldsymbol{\kappa}^{-1} \boldsymbol{\xi}), (n_k \boldsymbol{\Psi}^{-1} + \boldsymbol{\kappa}^{-1})^{-1}\big),$$

where $\hat{\boldsymbol{x}} = \frac{1}{n_k} \sum_{i:z_i=k}^{n} (\boldsymbol{x}_i - \boldsymbol{y}_i \boldsymbol{\Lambda}_k^T)$ for $k = 1, \ldots, K$.
For the variance of the noise in the model, $\boldsymbol{\Psi}$ (see Equation 2.5), we find it more convenient to work with the precision $\boldsymbol{\Psi}^{-1} = \text{diag}(\sigma_1^{-2}, \ldots, \sigma_p^{-2})$ and to define a matrix $S = \sum_{k=1}^{K} \sum_{i:z_i=k}^{n} (\boldsymbol{x}_i - \boldsymbol{y}_i \boldsymbol{\Lambda}_k^T - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{y}_i \boldsymbol{\Lambda}_k^T - \boldsymbol{\mu}_k)^T$. The Gamma full conditional posterior of each element on the diagonal is now given by:

$$\sigma_r^{-2} \sim \text{Ga}(\alpha + n/2, \tau + S_{rr}/2).$$

We use the zero mean Gaussian prior $\boldsymbol{\Lambda}_{kr} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Omega})$, for $k = 1, \ldots, K$ and $r = 1, \ldots, p$

which gives a Gaussian full conditional posterior:

$$\boldsymbol{\Lambda}_{kr} \sim \mathcal{N}\big((\boldsymbol{\Omega}^{-1} + \sigma_r^{-2}(\boldsymbol{Y}_k^T\boldsymbol{Y}_k))^{-1}(\sigma_r^{-2}\boldsymbol{Y}_k^T\bar{\boldsymbol{X}}_{kr}), (\boldsymbol{\Omega}^{-1} + \sigma_r^{-2}(\boldsymbol{Y}_k^T\boldsymbol{Y}_k))^{-1}\big),$$

where $\bar{\boldsymbol{X}}_{kr}$ is the data matrix obtained from $\bar{\boldsymbol{x}}_k = \boldsymbol{x} - \boldsymbol{\mu}_k$. We assume $\boldsymbol{\Omega}$ to be diagonal and find it more convenient to work with the precision $\boldsymbol{\Omega}^{-1} = \text{diag}(\omega_1^{-2}, \ldots, \omega_q^{-2})$ and to define $B = \sum_{k=1}^{K}\sum_{r=1}^{p}\boldsymbol{\Lambda}_{kr}\boldsymbol{\Lambda}_{kr}^T$. Finally the Gamma full conditional posterior for ω_c^{-2} is given by:

$$\omega_c^{-2} \sim \text{Ga}(g + Kp/2, h + B_{cc}/2), \qquad c = 1, \ldots, q.$$

3.2 Stochastic Model Selection

The previous section described how the MFA model could be estimated for fixed K and q using a Gibbs' sampler. We now consider the problem of inferring the correct value for K. This is a model selection problem. A description of the Bayesian approach to model selection was provided in Section 2.2.1 outlining the inherent penalisation for overly complex models which makes it preferable to the standard non-Bayesian alternatives.

In order to infer K it is necessary to compute the joint posterior over all models, $\mathcal{M}_k$, and associated model parameters, θ_k. However as we change the model order the dimensionality of θ_k naturally changes as well. The required joint posterior is analytically intractable and we therefore turn to methods of approximation of which there are two; stochastic approximation and variational approximation. The method we have focused on is the stochastic approach, using Birth-Death MCMC developed by Stephens in [125, 126] and applied to the MFA in [49], which is described in detail in the following section. Section 3.4 provides a comparative study evaluating the performance of the BDMCMC method with it's variational analog and to prepare for this a brief description of variational approximation, introduced earlier in Section 2.3.2, applied to the MFA is provided in Section 3.3. The next section describes the stochastic model selection method used.

3.2.1 Birth-Death MCMC for K

Stochastic model selection methods are MCMC algorithms capable of making trans-dimensional moves. This allows simulation from the posterior over model parameters and model index. The most widely used of these algorithms is the Reversible Jump MCMC (RJMCMC) [110]. An alternative method known as Birth-Death MCMC was developed by Stephens [125, 126] and applied to the MFA model in [48, 49]. BDMCMC is based on the theory of point processes and it has been shown to be a limit of RJMCMC where holding times, to be described later, replace accept or reject stages. An excellent comparison of RJMCMC and BDMCMC can be found in [20] where the authors demonstrate the many similarities between the methods. In the nested scheme we implement, performing model selection over the number of factor analysers and the number of factors in each analyser, the use of BDMCMC is more suitable due to it's greater portability and modularity. Within this section we will assume familiarity with the mixture model notation as used in Section 2.1.4.

The central idea behind this approach is to view each component of the mixture model as a marked point in the parameter space, where the marks correspond to the associated parameters, and adapt the methodology of point process simulation to help construct a Markov chain with the posterior distribution of the parameters as its equilibrium distribution [49]. For this method to hold we require that the posterior distribution satisfy two conditions; it is independent of the model labels and invariant to permutations of the parameters [1]. Studying the MFA formula:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}_p(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k \boldsymbol{\Lambda}_k^T + \boldsymbol{\Psi}),$$

it can be seen that the likelihood of a mixture model satisfies the invariance to labels requirement and parameter permutations. If prior distributions are chosen that are also invariant to labelling and permutations the posterior will have this quality. Later in Section 3.2.4 we will see that this invariance creates difficulties when computing the stochastic average, see Equation 2.21, due to a phenomenon known as label switching.

Within the birth-death framework births and deaths are defined, using the notation $/$ to denote a model without a specific component, as follows:

Births: If at time $t \in \mathcal{R}^+$ the process is at $m = \{(\pi_1, \theta_1), \ldots, (\pi_K, \theta_K)\} \in \chi_K$, where χ_K denotes the parameter space of the mixture model with K components, and a birth is said to occur at $(\pi_{K+1}, \theta_{K+1})$, then the process jumps to:

$$m \cup (\pi_{K+1}, \theta_{K+1}) = \{(\pi_1(1 - \pi_{K+1}), \theta_1), \ldots, (\pi_K(1 - \pi_{K+1}), \theta_K), (\pi_{K+1}, \theta_{K+1})\} \in \chi_{K+1}.$$

Deaths: If at time $t \in \mathcal{R}^+$ the process is at $m = \{(\pi_1, \theta_1), \ldots, (\pi_K, \theta_K)\} \in \chi_K$, and a death is said to occur at $(\pi_k, \theta_k) \in m$, then the process jumps to:

$$m/(\pi_k, \theta_k) = \left\{ \left(\frac{\pi_1}{1 - \pi_k}, \theta_1 \right), \ldots, \left(\frac{\pi_{k-1}}{1 - \pi_k}, \theta_{k-1} \right), \left(\frac{\pi_{k+1}}{1 - \pi_k}, \theta_{k+1} \right), \ldots, \left(\frac{\pi_K}{1 - \pi_k}, \theta_K \right) \right\} \in \chi_{K-1}.$$

Thus a birth increases the number of components by one, while a death decreases the number of components by one. These two operations are the inverse of each other and the constraint of $\pi_1 + \ldots, +\pi_K = 1$ is preserved by appropriate scaling of the mixing proportions after a birth or death. When a birth occurs a complete collection of component parameters are drawn from the prior distributions described in Section 3.1.1.

With births and deaths defined we now consider the continuous time Markov birth-death process, where births and deaths occur as independent Poisson processes. The rates at which these births and deaths occur is determined by the birth rate, $\beta(m)$ and death rate $\delta(m)$. The time to the next birth/death event is then exponentially distributed with mean $1/(\beta(m) + \delta(m))$ and the probability of it being a birth or death is:

$$p(\text{birth}) = \frac{\beta(m)}{\beta(m) + \delta(m)}, \quad p(\text{death}) = \frac{\delta(m)}{\beta(m) + \delta(m)}.$$

These rates determine the stationary distribution of the process [**?**]. If we set the birth rate, $\beta(m)$ to be a constant, λ_b, it has been proved that the process will have the correct stationary distribution if each point dies independently of each other as a Poisson process with rate [**?**]:

$$\delta_k(m) = \lambda_b \frac{(L(m \backslash (\pi_k, \theta_k))}{L(m)} \frac{p(K-1)}{Kp(K)},$$

where $L(m)$ is the likelihood evaluated for m using the mixture distribution formula and $p(K)$ is the prior probability on the number of components. Thus the death rate is calculated in such a way that components which do not fit the data well have a high death rate and are killed quickly. So the process constantly generates new births but reverses bad ones quickly. The total death rate, $\delta(m) = \sum_{k=1}^{K} \delta_k(m)$, is the sum of all component death rates.

The pseudo code for the Birth-Death process for K is shown in Algorithm 1, this is referred to as the naive birth-death process as it does not include the Gibbs update stage for the parameters, which instead remain at the values drawn from the priors. Starting with the initial model $m = \{(\pi_1, \theta_1), \ldots, (\pi_K, \theta_K)\}$ iterate the following steps:

Algorithm 1 Pseudo code for naive birth-death process.

1. Define the birth rate $\beta = \lambda_b$, hold time $\mathrm{T}_{\text{hold}} = 0$ and $\mathrm{T}_0 = 0$.

2. Calculate the death rate for each component, the death rate for component k being given by:

$$\delta_k(m) = \lambda_b \frac{(L(m \backslash (\pi_k, \theta_k))}{L(m)} \frac{p(K-1)}{Kp(K)} \qquad (k = 1, \ldots, K).$$

3. Calculate the total death rate $\delta(m) = \sum_{j=1}^{K} \delta_k(m)$.

4. Simulate the time to the next birth/death move, t_{move}, from an exponential distribution with mean $1/(\lambda_b + \delta(m))$.

5. Simulate the type of jump: birth or death with respective probabilities

$$p(\text{birth}) = \frac{\lambda_b}{\lambda_b + \delta(m)}, \quad p(\text{death}) = \frac{\delta(m)}{\lambda_b + \delta(m)}.$$

6. Adjust m to reflect the birth or death:
   Birth: Generate a new component by drawing all the associated parameters from their respective prior distributions and adjusting the mixing proportions appropriately. K becomes $K + 1$.

   Death: Select a component to die with probability $\delta_k(m)/\delta(m)$ for $k = 1, \ldots, K$. K becomes $K - 1$.

7. $\mathrm{T}_0 = \mathrm{T}_0 + \mathrm{t}_{\text{move.}}$

8. Return to step two until T_0 exceeds the hold time T_{hold}. When it exceeds the hold time, draw a sample of K^t and return to step one to simulate the value for K^{t+1}.

9. Repeat until $\mathrm{t} = \mathrm{T}_{\text{end}}$.

In Algorithm 1 we refer to a hold time, T_{hold}. This is the time that must be exceeded for the algorithm to exit the inner loop and take a sample. This idea is illustrated in Figure 3.3. Within each discreet sampling instance the continuous time naive algorithm runs for

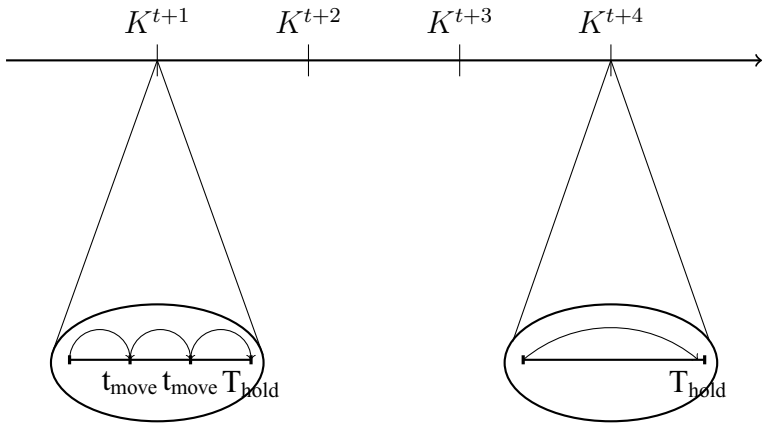


Figure 3.3: Illustration of the function of the hold time, T_{hold}, in the birth-death scheme: see text for details.

a set time T_{hold}. For each birth or death move there is a time to next event, t_{move}. The expected time to the next move is inversely proportional to the death rate so this hold time encourages the sampler to stay in regions of low death rate longer and draw more samples of K [49]. In poor configurations of the model t_{move} will be low as illustrated at sample K^{t+1} where within the hold time we get multiple birth/death events which all have a low t_{move}. In contrast at sampling instance K^{t+4} the last birth/death produced a good configuration for which t_{move} exceeds T_{hold} and we jump out of the hold time and sample this configuration without further moves. The algorithm requires the specification of λ_b, the constant birth rate, and the hold time, T_{hold} for which the birth-death process is run within samples. However doubling λ_b is mathematically equivalent to doubling T_{hold} and so we are free to fix $T_{hold} = 1$ and specify a value for λ_b. Larger values of λ_b will result in better mixing over K, at the cost of more computation *per iteration time* and it is not clear how an optimal balance between these factors could be achieved [**?**].

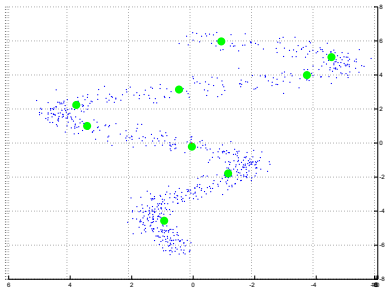
3.2.2 Naive Birth-Death Algorithm

To demonstrate the naive birth-death process, a selection of results are shown using the algorithm as described by the pseudo code in Algorithm 1 on some synthetic data. The data used in this example is the Ueda spiral data set [134], a 3-D spiral of 800 points subjected to some Gaussian noise shown in Figure 3.4. The naive birth-death algorithm does not update the posterior distributions of the parameters using the Gibbs sampler but only the value K, the number of clusters which are born with a single fixed factor. With no updates of the Gibbs sampler, what occurs is simply a stochastic search of the data space using random draws from the priors. Clusters born in areas of high likelihood are retained while those in low areas die. The computational cost of this naive algorithm is quite low, the simulation below took less than a minute to run 10,000 samples, and in terms of performance it can be seen in Figure 3.4, where the cluster centres are shown using a green marker, that it does a good job of finding good cluster positions in the data. This, as noted by [**?**], makes it an ideal way to initialise the full algorithm.

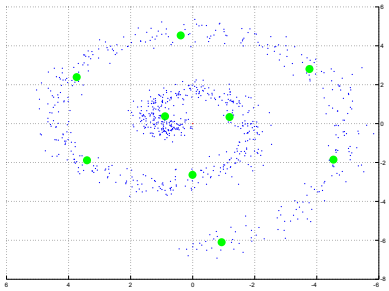
Figures 3.4a and 3.4b show the locations of the cluster centers, after 10,000 samples, relative to the noisy data. The arrangement observed is representative of the shape of the data and all of the data space has been well explored. In Figure 3.4c the MCMC sample path plot of K shows how initially just adding components improves the model but as the locations become better the number of components reduces to just those for which there is adequate support.

3.2.3 Estimating the Number of Factors

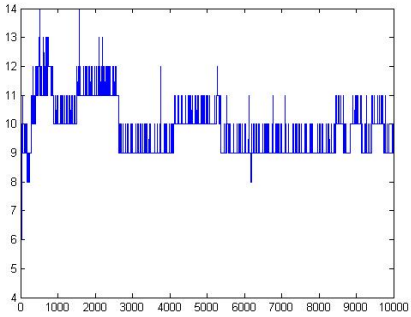
The BDMCMC scheme described in the previous section can also be used to make inference about the number of factors in each cluster of the MFA model. This gives each local factor analyser its own internal dimension q_k and we define the K-dimensional vector $\boldsymbol{q} = \{q_1, \ldots, q_K\}$ [48]. The complete algorithm is now implemented using a nested scheme, where first K and then $\boldsymbol{q}$ are simulated over. From some starting point the parameters are initialised to $\theta = \{K^t, \boldsymbol{q}^t, \boldsymbol{\pi}^t, \boldsymbol{\mu}^t, \boldsymbol{\Lambda}^t, \boldsymbol{\Psi}^t\}$ and allow K to vary for a fixed time t_0 and then sample K^{t+1}. The number of factors, $\boldsymbol{q}$, are then allowed to vary for every individual cluster $k = 1, \ldots, K^{t+1}$, for a fixed time t_0 using the parameters $\theta = \{K^{t+1}, \boldsymbol{q}^t, \boldsymbol{\pi}^t, \boldsymbol{\mu}^t, \boldsymbol{\Lambda}^t, \boldsymbol{\Psi}^t\}$



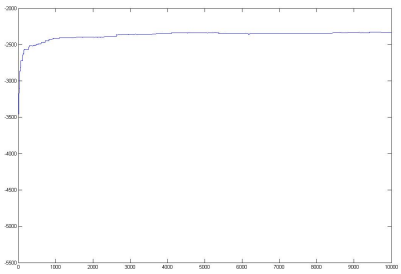
(a) Data in profile



(b) Data from above



(c) MCMC output for K



(d) Likelihood

Figure 3.4: MFA applied to Ueda spiral data set [134] using BDMCMC. (a) and (b): data, in profile and from above, plotted in blue with the means of the centres of the nine components in green. (c): samples path obtained for K. (d): likelihood which is improved over the run.

until we obtain $\boldsymbol{q}^{t+1}$. The Gibbs sampler described in Section 3.1.1 is then used to update the remaining parameters producing a new sample from the target distribution over all parameters. From these samples the quantities of interest may be estimated by appropriate sample path averages, for example:

$$\begin{aligned} p(K = i|\boldsymbol{x}) &= E(I(K = i|\boldsymbol{x})) \qquad \text{of } i = 1, \ldots, K_{\max}, \\ &\approx \frac{1}{n} \sum_{t=1}^{n} I(K^{(t)} = i). \end{aligned}$$

As part of the algorithm we need to specify priors on the values of K and q. In our simulations we used appropriately bounded uniform priors which place equal probability on all values of K and q within a certain range and are thus relatively uninformative.

3.2.4 Label Switching

The use of MCMC methods for mixture models requires a careful treatment of a phenomenon known as label switching. We will give a brief outline of the problem and the solution we have used here but for a complete overview of the topic see [70].

One of the main challenges of a Bayesian analysis using mixture models is the non-identifiability of the components [82]. That is, if exchangable priors are placed upon the parameters of a mixture model, then the resulting posterior distribution will be invariant to permutations in the labelling of the parameters. As a result, the marginal posterior distributions for the parameters will be identical for each mixture component. Therefore, during MCMC simulation, the sampler encounters the symmetries of the posterior distribution and the interpretation of the labels switch. It is then meaningless to draw inference directly from the MCMC output using sample path averaging [70]. Label switching significantly increases the effort required to produce a satisfactory Bayesian analysis of the data, but is a prerequisite of convergence of an MCMC sampler and therefore must be addressed.

There are three main approaches to resolving the labels switching issue: identifiability constraints (IC) [31], relabelling algorithms [**?**] and label invariant loss functions [22]. The method we have chosen to use is a pivot relabelling algorithm developed by [100] which matches all samples in the path to a pivot permutation of the labels. This very recent algorithm is both simple and efficient, and has the advantage that the computational cost depends on the length of the simulated chain but not on the parameter space dimension. In contrast IC based algorithms are unsuitable for use in multivariate situations while the decision theoretic approach of a label invariant loss function is quite complicated and computationally very intensive [100].

The form of a relabelling algorithm is exactly that of a k-means clustering algorithm [70]. Under an inferential objective the permutations of the labelling are induced or discovered so that all of the samples are labelled in the same way. The pivot relabelling algorithm we have chosen operates in the allocation space, i.e. $\boldsymbol{z}$ or the cluster indexes for each point. It seeks to minimise a matching distance between the allocation variables for each point at each iteration and a specified pivot orientation of the labels, typically chosen as the maximum a posteriori estimate [100]. The labels are permuted so as to minimise this matching

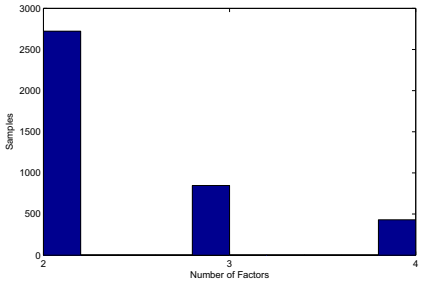
distance and label all the samples to the template permutation. The algorithm can only be deployed once the simulation has completed, for fixed K, and this requires that all the sampled parameters are stored as well as the labels and also specify the number of clusters post simulation. This then allows us to make sensible estimates of $\boldsymbol{\mu}, \boldsymbol{\pi}$ and $\boldsymbol{\Psi}$ over the sample path.

This relabelling algorithm combined with the BDMCMC methods discussed gives us a complete method for conducting Bayesian analysis of the MFA model for unknown $K$and q_k. In the next section we conduct a set of simple verification tests to demonstrate the correct working of the algorithm to prepare for its use in more demanding challenges.

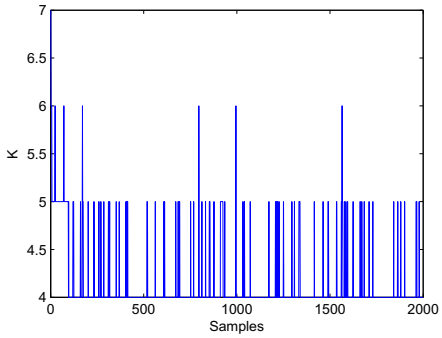
3.2.5 Verification Tests

The MFA model with variable K and q_k estimated by BDMCMC which we have just described is a complex algorithm. The full algorithm will be deployed in the analysis of real data in Section 3.5 and also in a comparison study with an alternative variational algorithm in Section 3.4. To prepare for these we first verify that the algorithm is working correctly and conduct a number of tests as described in [48] examining each aspect of the algorithm in turn. Synthetic data was generated from the factor model, Equation 2.4 and three tests conducted. First data composed of a single cluster with $p = 10$ and $q = 2$ was used to test how accurately the model could estimate the latent dimensionality. Then a data set with $K = 4$, $q_k = q = 1$, and $p = 3$ was tested to assess the algorithm's ability to cluster data accurately. Finally a data set with $K = 2$, $p = 10$ and $q_k = [1, 5]$ was tested to see how well the algorithm could perform both operations simultaneously. Selected results are shown in Figure 3.5 illustrating the good performance of the algorithm. Performance is judged on the posterior estimates for K and q_k and whether or not they approach the truth.

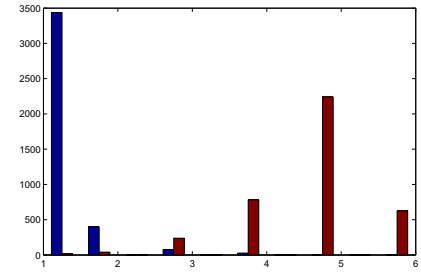
As a more complicated clustering problem, a data set consisting of two overlapping clusters was generated with $p = 3$ and $q_k = q = 1$. A plot showing the original two clusters and the reconstructed points obtained is shown in Figure 3.5 and it can be seen that the algorithm is able to identify clusters even when their mean vectors are very close and there is substantial overlap of points.



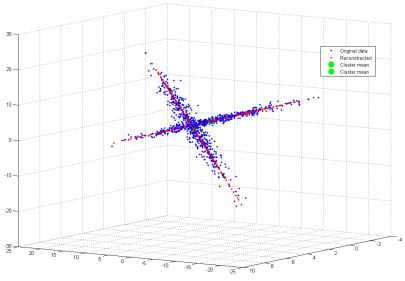
(a) Posterior distribution of q



(b) Estimation of K



(c) Posterior distribution of q_k



(d) Two clusters that cross with $q = 1$

Figure 3.5: Results from a set of verification tests: see text for details. (a): posterior distribution of q for data with $p = 10$, $q = 2$ and $K = 1$, peaked at the correct value. (b): samples obtained for the posterior distribution of K, for data with $p = 3$, $q_k = q = 1$ and $K = 4$. (c): histogram of samples obtained for the posterior distribution of q_k, for the $K = 2$, $p = 10$ and $q_k = \{1, 5\}$ test data. (d): two overlapping clusters were generated with $q = 1$ as a more difficult clustering problem. The model correct identifies K and q_k and the noisy data is plotted in blue with the reconstructed single factor 'signal' in red.

3.3 Variational Bayesian Mixtures of Factor Analysers

Section 2.3.2 gave an introduction to the theory of variational Bayesian approximations. This method has been used by Beal and Ghahramani [52, 10] as an alternative means of conducting Bayesian inference of the MFA model. Section 3.4 describes an empirical comparison of the variational Bayesian approximation and BDMCMC investigating their different characteristics and capabilities and to prepare for this the variational approach to the MFA is discussed here. The variational method is a complicated procedure and we do not explore it in great detail. For a full explanation of the variational solution for MFA see

[52, 10] but broadly speaking it works in the following manner.

The variational approximation places a lower bound on the model evidence, discussed in Section 2.2.1, using Jensen's inequality:

$$\ln p(\boldsymbol{x}) = \ln \int p(\boldsymbol{x}, \theta) \mathrm{d}\theta \geq \int q(\theta) \ln \frac{p(\boldsymbol{x}, \theta)}{q(\theta)} \mathrm{d}\theta \equiv \mathcal{F},$$

which we seek to maximise [52]. Maximising $\mathcal{F}$ is equivalent to minimising the KL-divergence, an entropy measure, between $q(\theta)$, a tractable proposal distribution which we use to approximate the posterior $p(\theta|\boldsymbol{x})$. For the proposal distribution, $q(\theta)$, we use a factorised approximation to the joint posterior over all parameters. Due to the conditional independence structure of the Bayesian MFA model the only approximation that is needed apart from that between the latent variables and the model parameters is the factorisation $q(\boldsymbol{v}, \boldsymbol{\Lambda}, \boldsymbol{\mu}) \approx q(\boldsymbol{v})q(\boldsymbol{\Lambda})q(\boldsymbol{\mu})$ where $\boldsymbol{v}$ is the precision over $\boldsymbol{\Lambda}$.

Model exploration in the variational approach again occurs in single increment birth/death moves, however the heuristics for these moves are completely different to those of the BDMCMC. Component births do not occur spontaneously as previously. Instead, whenever $\mathcal{F}$ is judged to have stabilised a component is selected probabilistically based on how well it is performing and split. Dependent on the effect on $\mathcal{F}$ this birth can be accepted or rejected. Deaths occur naturally whenever a component is judged to have zero responsibility for the data [10].

In order to estimate the local dimensionality of each FA an Automatic Relevance Determination (ARD) [83] mechanism is used. Each factor analyser's dimensionality is set to the maximum possible and we use priors that discourage large factor loadings. The width of each prior is controlled by a hyperparameter and the result of learning with this method is that only those factors that are required remain active after learning [10]. Each column of each factor loading matrix is governed by a separate precision parameter, v_{lk}, $l = 1, \ldots, q$. If one of these precisions $v_{lk} \to \infty$ then the weights for the lth factor in the kth analyser will have to be very close to zero in order to maintain a high likelihood under this prior and this in turn leads the analyser to ignore this factor allowing the model to reduce the intrinsic dimensionality in the locale of that analyser [52].

This variational approach outlined above gives us another method for inferring the num-

ber of components, K, and intrinsic dimensionalities, q_k, for the MFA model. The approximations that arise in the BDMCMC outlined in Section 3.2 are due to the use of finite computing times, however if the sampler is run to convergence then it is essentially inferring the exact distributions required. In contract the variational approach approximates these distributions with simpler factorised assumptions. We therefore might expect to obtain different results between the two different methods in certain circumstances, for example it is a general result that a factorised variational approximation tends to give approximations to the posterior distribution that are too compact [52]. The variational methods has also been found to be sensitive to correlations in data [21] which could be an issue given the FA decomposition of the data covariance matrix into variance and covariance structure. An empirical comparison of both methods in this setting has not to our knowledge been performed to establish under what circumstances one method may fare better than the other. The form of the model used in the variational scheme is slightly different to the one proposed for the stochastic scheme, this is illustrated in Figure 3.6. A deterministic solution

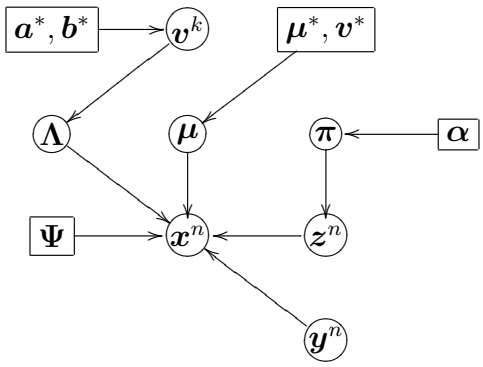


Figure 3.6: Directed Acyclic Graph for the formulation of the MFA used in the Variational Bayes approach. The main difference is that in the BDMCMC framework $\boldsymbol{\Psi}$ is a random variable.

for the variance of the noise matrix $\boldsymbol{\Psi}$ is used as opposed to the full posterior. This is conditional on the model parameters and is updated accordingly at each iteration of the algorithm. Correct estimation of the noise matrix is vital for correctly inferring the dimensionality. These differences between the two methods may produce different properties making them more suitable for different situations, such as in higher dimensional data or when data is sparser, and this behaviour will be investigated in the next section.

3.4 Comparative Study

The inference problems associated with the Bayesian approach to model selection are typically resolved by the application of either MCMC or variational methods. Chapter 2 discussed the theory behind both approaches and in this section we conduct an empirical evaluation of the performance obtained from each method in a variety of different scenarios using synthetic data.

It is well known that the computational costs of MCMC methods are high, particularly in data spaces of large dimension, and a great deal of research has been devoted over the years to developing more efficient sampling methods. In contrast variational methods are orders of magnitude faster but variational estimates of posterior distributions can be inaccurate. For example, they are often too concentrated [52]. MCMC methods however are guaranteed to converge to the 'true' posterior when run for sufficiently long. From a practical point of view it would be of great interest to understand under what circumstances variational methods could produce comparable or equivalent performance to MCMC methods. A recent study has sought to answer this question in the setting of Bayesian variable selection for linear regression [21]. We build on this work and extend it to the unsupervised setting in the following experiments. From a more theoretical perspective it has also been commented that another potential draw to developing a greater understanding of the variational approach is that it would represent a 'more' Bayesian solution to the problem rather than resorting to what are essentially frequentist sampling methods [10], but this is perhaps a touch dogmatic.

3.4.1 Methodology

The objective in these experiments is to correctly infer the number of clusters, K, and the number of factors per cluster q_k present in the data for a range of settings. We investigate performance with respect to two factors; number of data points per cluster, $n_k \in \{10, 25, 50, 100, 150\}$, and the difficulty of the clustering problem, which we increase by reducing the distance between clusters from 100% to 20% of its starting value in steps of 20%. This gives 25 different combinations to test. These 25 simulations are then repeated 10 times and averaged. There is no reason to expect to obtain different results if the MCMC

algorithm has converged and we find that the variance between repeated experiments is minor. The variational method can become 'trapped' in local maxima and terminate early and repetitions help reduce the effect of this. The effect of increasing the dimensionality of the data is also considered and the complete set of experiments is carried out for three settings $p \in \{10, 20, 30\}$. This gives us 750 individual simulations, per method.

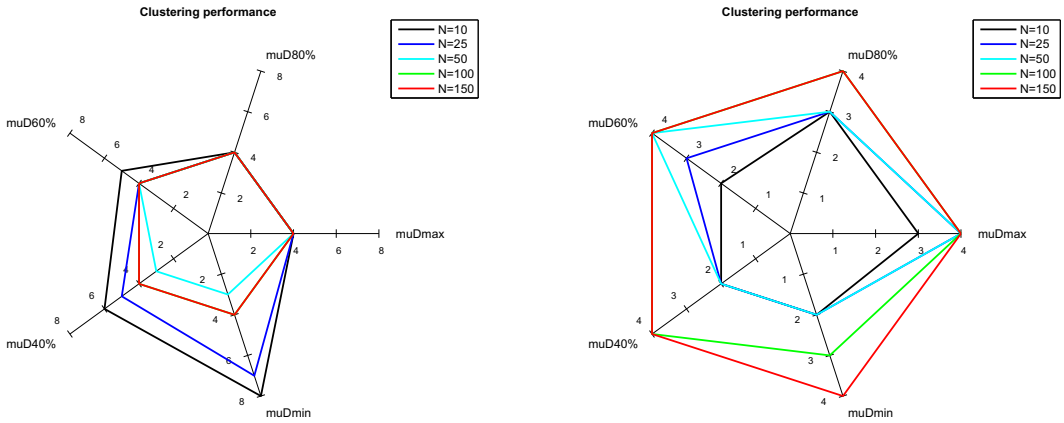
The final factor considered which may be relevant to performance is the robustness of each algorithm to data that deviates from the model assumptions. To test this the previous experiments are repeated but this time the data is subjected to additive noise from a Uniform distribution rather than Gaussian, as per the model assumptions. The results presented are the product of weeks of simulations and the variance between repeated simulations is minimal suggesting that further simulation would not significantly alter them in any way.

3.4.2 Data

The data used in the experiments consists of four clusters, the centres of which would be located at the corners of the hyper-cube with sides of length five. The clusters conform to the FA model as described in Section 2.1.1 with each having a different number of factors, $\boldsymbol{q} = [1, 2, 3, 4]$ and the data is a MFA described by Equation 3.1 with variable q_k. The $\boldsymbol{\Lambda}_k$ were formed from independent draws from the distribution $p(\boldsymbol{\Lambda}_{kr}) \sim \mathcal{N}(\mathbf{0}_q, \boldsymbol{I}_q)$, in keeping with the basic assumptions outlined in 3.1.1. The noise matrix $\boldsymbol{\Psi}$ was varied as either being from a $\mathcal{N}(\mathbf{0}_p, \boldsymbol{I}_p)$ or a uniform distribution $U(-3, 3)$ depending on the experimental set up.

3.4.3 Results

Given the large quantity of information to be summarised, spider plots [23] are utilised as a convenient means of illustrating the performance obtained in the tests determining K. The five axes are the distances between clusters, and the number of points per cluster are the five different coloured lines in the plot. Figure 3.7 shows the results obtained for $p = 10$ using data sampled from the model. The performance obtained in estimating the number of factors, $\boldsymbol{q} = [1, 2, 3, 4]$ is summarised by calculating the string edit distance [57] between $\boldsymbol{q}$ obtained and the true $\boldsymbol{q}$. This is limited to comparing to the true K, so if a method finds that there are two clusters, we compare the edit distance with zeros substituted for the missing



(a) MCMC spider plot for K, $p = 10$ (b) VB spider plot for K, $p = 10$

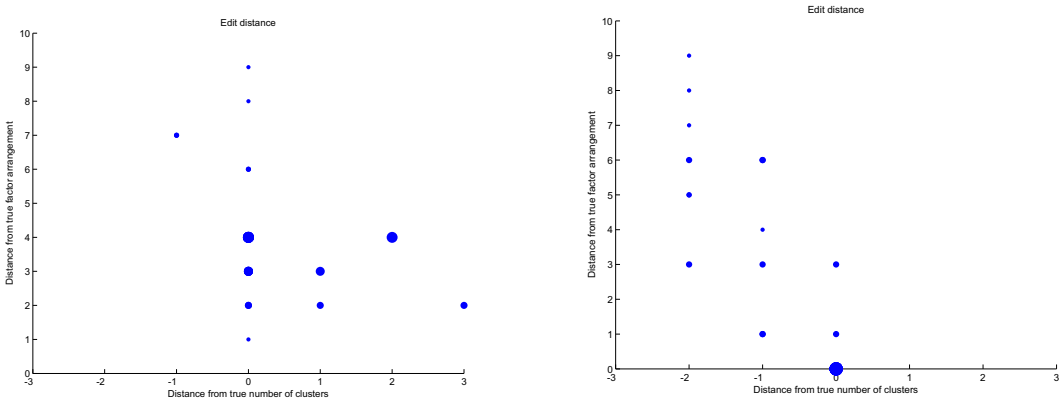
Figure 3.7: Results obtained from estimation of K varying the numbers of points per cluster and the distance between clusters, using the MCMC (a) and variational (b) methods for estimation. Comparison of the plots reveals the different behaviour of both methods with the number of points per cluster having a much greater impact on the performance of the variational method than on the MCMC.

clusters. In the case of a $K > K_{\text{true}}$ being determined we only consider four clusters. As the factor and cluster data are related we plot the value of the string edit distance obtained against the difference between K and K_{true}. Figure 3.8 shows the results obtained for $p = 10$ using data from the model.

Spider and factor plots for each setting of $p = \{10, 20, 30\}$ were produced for both the variational and MCMC methods, under the model and non-model data. We only show the results for $p = 10$ here with the remainder provided in Appendix A. The results are discussed in the next section.

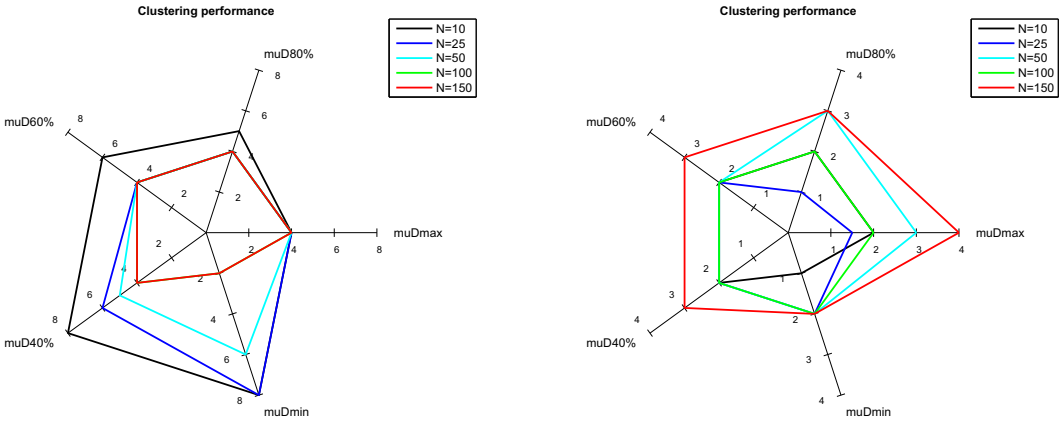
3.4.4 Analysis

The results of the simulation study for $p = 10$ are shown in Figures 3.7 - 3.10. The results for the remaining experiments are included in the Appendix A but here we analyse the main features observed over all experiments.



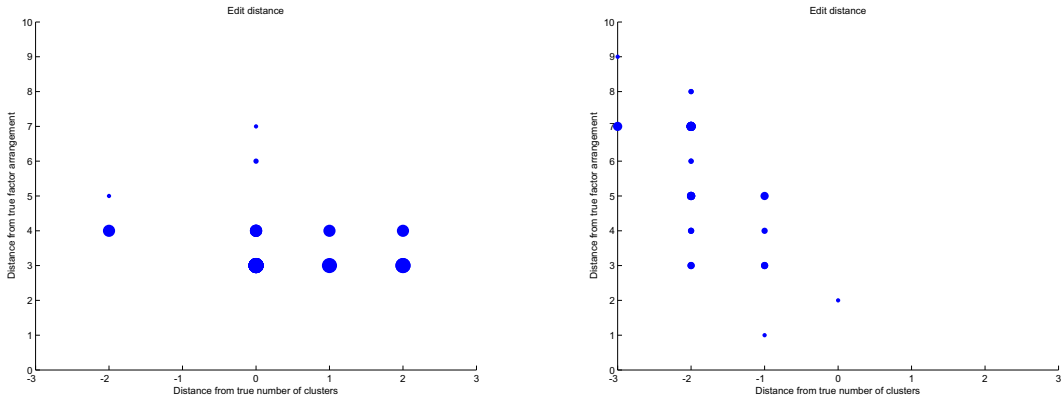
(a) MCMC factors, $p = 10$ (b) VB factors, $p = 10$

Figure 3.8: Results obtained for the comparison of the string edit distance between the true $\boldsymbol{q}$ and that estimated by the MCMC (a) and variational (b) method. Each plot consists of 25 results and the size of the data point reflects multiple occurrences of the same integer value. The x-axis shows the distance from true K with the origin representing correct estimation of both $\boldsymbol{q}$ and K and negative distance implying underestimation.



(a) MCMC spider plot, $p = 10$, uniform noise. (b) VB spider plot, $p = 10$, uniform noise.

Figure 3.9: Results obtained for estimation of K when data deviates from model assumptions i.e. is subject to uniform noise using the MCMC (a) and variational (b) methods. The same trends are observed as in 3.7 however performance degrades more rapidly with increasing difficulty of the experiment and the variational method suffers more than the MCMC.



(a) MCMC factors, $p = 10$, uniform noise.

(b) VB factors, $p = 10$, uniform noise.

Figure 3.10: Results obtained for estimation of $\boldsymbol{q}$ when data deviates from model assumptions i.e. is subject to uniform noise. using MCMC (a) and variational (b) methods for model estimation. Each plot consists of 25 results and the size of the data point reflects multiple occurrences of the same integer value. The x-axis shows the distance from true K with the origin representing correct estimation of both $\boldsymbol{q}$ and K.

Clustering Performance

The results obtained for the inference of K show an interesting trend that persists throughout all values of p. Comparison of the spider plots in Figures 3.7a and 3.7b highlight a key difference between the methods. The correct value of K is estimated by both methods when n_k is at its maximum value regardless of the difficulty of the clustering problem. However as n_k is reduced the performance observed changes very differently for either method. Focussing on Figure 3.7a it can be seen that for the first three axes, starting from the axis labelled 'muDmax', which represents the maximum distance between the cluster means, and going anti-clockwise, the correct value of K is inferred for all n_k apart from the minimum, $n_k = 10$. The subsequent axes show a range of different values for K. This implies that for the BDMCMC it is not the number of points per cluster that is the key factor but rather the difficulty of the clustering problem, as this increases, by reducing the distance between the centres, results deviate more from the true value.

In contrast, if we consider the results for the variational method shown in Figure 3.7b different behaviour is observed. While again for $n_k = 150$ the correct K is inferred at all distances, as the number of points is reduced below $n_k = 100$ the correct value of K is not inferred even at large distances between the centres. This implies that for the variational

method, there is a greater dependence on the number of points per cluster than for the MCMC method.

Another interesting feature observed is the tendency of the variational method to almost exclusively underestimate the value of K when it is incorrect. In the MFA the problem of inferring K is intrinsically linked to the problem of estimating q_k and the next section discusses this point further.

Factor performance

As we have previously commented the inference of K and $\boldsymbol{q}$ are coupled, therefore when trying to convey the results of the estimation of $\boldsymbol{q}$ we plot the string edit distance against the distance from the true value of K obtained. The tendency of the variational method to underestimate the value of K can be seen in Figure 3.8b which is skewed to the left of true K. This is linked to incorrect factor estimation as we see in cases when $\boldsymbol{q}$ is correct we also infer K correctly, the origin of the plot. These results correspond to the high values of n_k observed in the corresponding spider plot 3.7b but also provide us with insight as to what is happening in the underestimation of K, the string edit distance in these cases is quite high indicating the number of factors has been over estimated to a large extent. This could be due to the variational method's oversensitivity to correlations in data identified in [21] resulting in over estimation of the number of factors which in turn leads to under estimation of K. In contrast to this we see that the MCMC results are robust to incorrect estimation of $\boldsymbol{q}$ with a wide spread of results lying on the vertical zero axis.

Robustness to departures from the model assumptions

Figures 3.9 and 3.10 show the results of the experiments under conditions that deviate from the model assumptions. Specifically, the synthetic data is generated as:

$$\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{z}\boldsymbol{\Lambda}^T + \boldsymbol{\epsilon},$$

where:

$$\boldsymbol{\epsilon} = U(-3, 3).$$

The noise the factor model is subjected to no longer follows a Gaussian distribution as per the model assumptions but is drawn from a Uniform distribution, $U(-3, 3)$. The results obtained exhibit the main characteristics observed in the previous experiments however performance is seen to degrade more rapidly, in comparison to the previous experiment, with incorrect values for K inferred even with n_k at its maximum. The performance of the variational method suffers more with incorrect values of K estimated at all but the simplest setting, $n_k = 150$ with maximum distance between the clusters. The results obtained using MCMC are more robust to this departure from the model assumptions and correct values of K are inferred for all $n_k = \{150, 100, 50, 25\}$ as the distance between clusters is reduced from its maximum to 60% of this value. After this point the results obtained for K deteriorate and this is observed for the higher dimensions, the results of which are included in Appendix A. We conclude that the MCMC method is more robust. It should be noted that as observed in the previous experiments in situations where the variational method has incorrectly estimated K it exclusively underestimates the value of K. Figure 3.10b shows the combined results of the estimation of $\boldsymbol{q}$ and K and again the results are skewed to the left as in Figure 3.8b.

3.5 Analysis of ADI-R

The first half of this chapter has been dedicated to an exploration of the MFA model and the techniques required to perform Bayesian model selection over the number of clusters and their associated latent dimensionality in an unsupervised manner. We conclude this chapter with the application of these methods to a data set consisting of diagnostic information for a cohort of autistic subjects. Autism is known to consist of a heterogeneous mix of subjects and we use the MFA to detect the presence of sub-populations within an autism data set. To begin a brief introduction to the field of Autism Spectrum Disorders (ASDs) is provided in Section 3.5.1. Autism is a complex and broad condition and a full description would be beyond the scope of this thesis, instead we focus on the aspects relevant to our analysis. The data was gathered from the Autism Diagnostic Interview - Revised (ADI-R) which is described in Section 3.5.2. Previous research conducted on the same type of data is also discussed. We outline how these methods motivate the application of the MFA

model which as far as we are aware has never been performed. The data set used consists of the behavioural attributes of a group of 625 autistic subjects and is described in detail in Section 3.5.3. The results of the analysis are presented in Section 3.5.5 followed by a discussion in Section 3.5.9.

3.5.1 Autism Spectrum Disorders

The term Autism Spectrum Disorders (ASDs) is used to describe the group of pervasive developmental disorders characterised by qualitative abnormalities in reciprocal social interactions and patterns of communication, and by a restricted, stereotyped, repetitive repertoire of interests and activities [33]. ASDs are developmental disabilities resulting from dysfunction of the central nervous system and usually manifest before three years of age [66]. The expression of these deficits encompasses a broad spectrum of individuals ranging from idiopathic autism, the most severe form, to Asperger's syndrome, a milder form of autism in which individuals have relatively normal intelligence but experience great difficulty with social interactions. This heterogeneity in the ASD population makes the analysis of large samples of subjects a difficult issue and it is thought to be one explanation for the difficulty in pinpointing genes involved in autism. Often researchers go to great lengths to minimise this heterogeneity in their study samples by carefully selecting their subjects. An alternative approach to this is to cluster the sample into more homogeneous groups based on the behavioural abnormalities observed. The ADI-R provides a rich source of behavioural information which allows such analysis.

3.5.2 The Autism Diagnostic Interview - Revised

The ADI-R is a semi structured interview, conducted by a clinician with the parents of a subject, designed to differentiate those with autism from those with language impairments and intellectual disability [81]. It is generally accepted by the field as a valuable assessment instrument [120].

In order to make a diagnosis the ADI-R is used to quantify the abnormal behaviour displayed by the subject in the key domains of autism; social interaction, communication and repetitive and restricted interests. At the base or *item* level there is the raw data which

consists of responses to individual questions or criteria, the order of 100 of these per individual. These item scores are then transformed to generate 12 *sub-domain* scores. Each sub-domain has an interpretable meaning, for example providing a measure of a subject's 'failure to develop peer relationships'. The sub-domains are related to the three major domains of autism symptoms (social interaction, communication and repetitive and restricted behaviour). The scores of specific sub-domains are combined to give scores at the *domain* level, if these scores are above specified 'cut-off values' a subject is diagnosed as autistic [50], this is explained in further detail in Section 3.5.3.

The ADI-R provides a rich source of behavioural information for analysis. Given the nature of the data set, measures of behavioural abnormalities linked to an underlying neurological disorder, we believe the latent variable model of FA to be an appropriate method.

A key assumption of the FA model is that the data is continuous in nature. At the raw level the ADI-R data is categorical and this assumption is not valid. Therefore analysis is restricted to the domain and sub-domain levels. These are summary scores of the categorical data that span a much broader range and for the purposes of the analysis will be treated as continuous. Previous research has also used FA in this way and a brief overview of some of this work is provided in the next section.

The fact that the ASD population is known to be a heterogeneous mix of subjects who exhibit widely varying differences in symptoms demands the use of a class of models that can capture such interactions, the fitting of a global model encompassing all subjects would be a mistake. A mixture of factor analysers allows clustering of this population into homogeneous groups while simultaneously carrying out local latent structure analysis.

Understanding the structure of autism symptoms can move the field forward in three important ways: it can improve our diagnostic and classification systems, provide valuable information for genetic studies [120] as well as giving clues to the underlying differences in manifestations of what is a wide and varied condition.

Research Involving the ADI-R

In the past there has been a large volume of work done analysing and manipulating large sets of ADI-R data [120, 93, 123, 26, 66, 122, 30, 50]. Research efforts in this area gen-

erally fall into two categories. There are numerous groups using dimensionality reduction techniques for exploratory analysis of the structure of the ADI-R with the chief aim of assessing and improving it's capabilities to accurately diagnose individuals [120, 50, 30]. Other groups have used the ADI-R as a means of clustering a heterogeneous population of subjects into homogeneous subgroups for whom more coherent analysis can be made [66, 93, 122, 123]. These works are discussed in detail here.

Recent work has utilised the ADI-R as a means of 'stratifying' the heterogeneous population into homogeneous groups with similar characteristics in terms of behaviour [66, 93, 122, 123]. It is supposed that genetic analysis of these clusters will be much more informative and suffer from less noise. Early work consisted of simply selecting one *item* of the ADI-R, for example developmental regression [93] or language development [122], and segregating patients as to the level of this feature. Most recently multivariate clustering approaches have been employed to group patients into predetermined numbers of clusters based on similarity of symptoms [66]. A feature that has emerged from this vein of research is that of a 'severity gradient'[123] as an underlying feature which controls the allocation of patients to each cluster. Another paper found the use of non-probabilistic clustering methods produced four distinct classes of subjects, two characterised by varying degrees of language difficulties, one by the prevalence of savant skills, and another as being of intermediate severity [66]. The need to specify the number of clusters *a priori* is a major drawback of the clustering research described. The use of more sophisticated probabilistic clustering methods is an improvement made by our approach.

Studies using the ADI-R to examine the structure of autism symptoms have been conducted at the item [66, 93, 122, 30, 120, 26] and sub-domain levels [50, 123]. In the main these studies have used PCA to reduce the data to fewer dimensions which account for most of the variance in the data [30, 66, 93, 123, 50].

The two key themes which emerged from a review of the autism literature are that it is generally accepted that decomposing the heterogeneous population into homogeneous clusters aids analysis. Also the analysis of ADI-R data can reveal important differences in patients' behavioural characteristics which are of interest to the community in exploring the autism phenotype. The work presented here combines elements of both these themes, similar to the genetic studies we utilise the ADI-R domain and sub-domain scores to cluster,

in an unsupervised manner, a population of subjects. We also carry out factor analysis to gain insight about the latent structure of each cluster individually. This combination of clustering and lataent structure analysis is a novel framework for the analysis of the ADI-R.

3.5.3 The Data

The data analysed was collected as part of research conducted by the International Molecular Genetic Study of Autism Consortium (IMGSAC) [98, 97]. IMGSAC specifically selected subjects from multiplex families (families with multiple occurrences of autism) in an effort to identify underlying genetic factors. Thus the data set used does not represent a sample from the general autism population but rather the $10-15\%$ of instances where more than one child in the family has autism. The data consists of 625 ADI-R interviews in total for individual subjects from 299 different families. The family arrangement is generally pairs of siblings, but also features families with three and one instance of a family with four affected subjects. From question ratings, the scores were formed using the standard ADI-R algorithm. This included setting cut-offs of 10, 8, 7 and 3 respectively for social, verbal, non-verbal and repetitive behaviour domains. For a diagnosis of autism to be made a subject must score above these cut-off scores in every category [81]. Analysis was conducted on the domain and sub-domain summary scores of the ADI-R. The sub-domains are obtained by summing specific item level questions relating to the same type of behaviour and are defined in Table 3.1.
In turn, the domain scores are the sum-scores of three to four question items that are qualitative measures of impairments in various sub domains:

- *Social*: ADI-R Social Interaction score S (= S1 + S2 + S3 + S4)
- *Verbal*: ADI-R Verbal Communication C (= C1 + C4 + C2V + C3V)
- *Non-Verbal*: ADI-R Non-Verbal Communication score (= C1 +C4)
- *Repetitive Behaviour*: ADI-R Repetitive Behaviour R (= R1 +R2 + R3 + R4)

S1	failure to use non-verbal behaviour to regulate social interaction
S2	failure to develop peer relationships
S3	lack of shared enjoyment
S4	lack of socioemotional reciprocity
C1	delay in spoken language and failure to compensate through gesture
C2V	relative failure to initiate or sustain conversational interchange
C3V	stereotyped, repetitive or idiosyncratic speech
C4	lack of spontaneous make believe or social imitative play
R1	encompassing preoccupation or circumscribed interests
R2	apparently compulsive adherence to non-functional routines or rituals
R3	stereotyped and repetitive motor mannerisms
R4	preoccupation with parts of objects or non-functional elements of materials

Table 3.1: Sub-domains of ADI-R [50]

As part of initial analysis a group of seven outliers were uncovered. These were found to be patients without verbal skills and are noted among the literature as being potential confounding patients who's age makes them unsuitable candidates for the diagnostic interview.

3.5.4 The Rand Index

In the analysis we perform it will be necessary to compare clusterings or partitionings of the data, for example between the domain and sub domain levels, to demonstrate consistency. An objective performance measure for such a comparison is the Rand index [109]. The Rand index is calculated as:

$$R = \frac{a+b}{a+b+c+d} \tag{3.2}$$

where:

- a, is the number of pairs of elements that are in the same cluster in both partitions
- b, is the number of pairs of elements that are in different clusters in both partitions
- c, is the number of pairs of elements that are in the same cluster in partition one and in different clusters in partition two

- d, is the number of pairs of elements that are in different clusters in partition one and in the same cluster in partition two.

The Rand Index has a value between zero and one, with zero indicating two partitions totally disagree or one indicating they are exactly the same. We also use the Adjusted Rand Index which is the Rand Index corrected for chance. This takes values in the range -1 to 1 where a score of 0 indicates the agreement obtained is no greater than what would be expected by chance.

3.5.5 Results

In this section we present the results of the application of the MFA algorithm to both the 4-D domain data and the 12-D subdomain data.

Domain Data

The limited number of variables at the domain level only allows us to reliably estimate a single factor solution. With $q_k = 1$ the BDMCMC algorithm was used to estimate the number of clusters in the data using the MFA model. The algorithm returned a distribution over the number of clusters that was peaked around $K = 5$ but also placed high probability on a $K = 6$ solution. The means of the posterior means and mixing proportions of each cluster are shown in Table 3.2 and Table 3.3 for typical instances of both arrangements.

Cluster	Social	Verbal	NonV	Rep	Population
1	27.3	15.5	13.36	8.24	33%
2	26.6	11.4	12.9	6.2	23%
3	24.2	10.0	9.7	5.7	23%
4	15.9	9.1	7.7	5.4	19%
5	1.5	3.1	0.2	5.1	1%

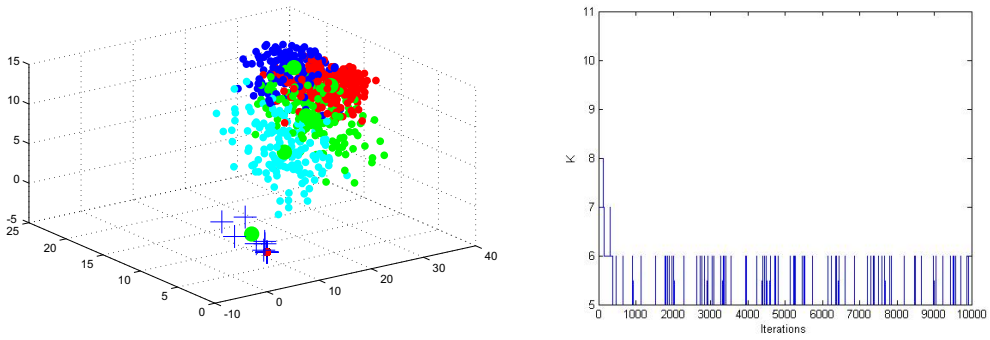
Table 3.2: Posterior means of the mean and relative sizes of each cluster for the $K = 5$ solution.

Cluster	Social	Verbal	NonV	Rep	Population
1	28.0	15.2	14.0	9.0	29%
2	27.4	12.1	12.7	6.5	20%
3	25.2	11.8	11.6	6.4	19%
4	22.43	10.0	10.7	5.3	16%
5	15.7	8.6	7.2	5.1	15%
6	1.5	3.4	0.7	4.6	1%

Table 3.3: Posterior means of the mean and relative sizes of each cluster for the $K = 6$ solution.

The two partitionings were compared using the Rand index. The Rand index obtained was .75 indicating that many of the points clustered together in one partition are also clustered together in the second partition. The Adjusted Rand index obtained was .28, the adjusted Rand index takes values from -1 to 1 with zero indicating that the agreement is exactly what would be expected just by chance. The positive score indicates that the agreement between partitions is statistically significant. Examining the results for the means of the clusters in Tables 3.2 and 3.3 it can be seen that the clusters are ordered in increasing severity of symptoms. The relative sizes of the clusters also reveal that the sample population is skewed towards subjects displaying a high level of disorder. Comparing the $K = 5$ and $K = 6$ solutions it appears that the extra cluster in the $K = 6$ clustering arises due to a further decomposition of the three largest clusters in the $K = 5$ solution with the other two clusters occurring almost identically in both partitions as clusters 4 and 5 in the first and clusters 5 and 6 in the second partition. Given the greater probability placed on $K = 5$ we choose to proceed in our analysis with this solution but we note that the possibility of further decomposition of the three largest clusters.

The scores of the subjects in the IMGSAC data set for the first three domains (social interaction, verbal and non-verbal communication skills) are plotted in Figure 3.11 with the data points colour coded according to their cluster membership. The loading matrices for the first four clusters in Table 3.2 are shown as Hinton diagrams [61] in Figure 3.12. Given the distribution of the clusters observed in Figure 3.11 it could be supposed that the single factor in each loading matrix could be thought of as a 'severity factor'. It is interesting that the loading matrices for each group are markedly different indicating that this severity



(a) Clustered domain data $K = 5$ (b) Output K

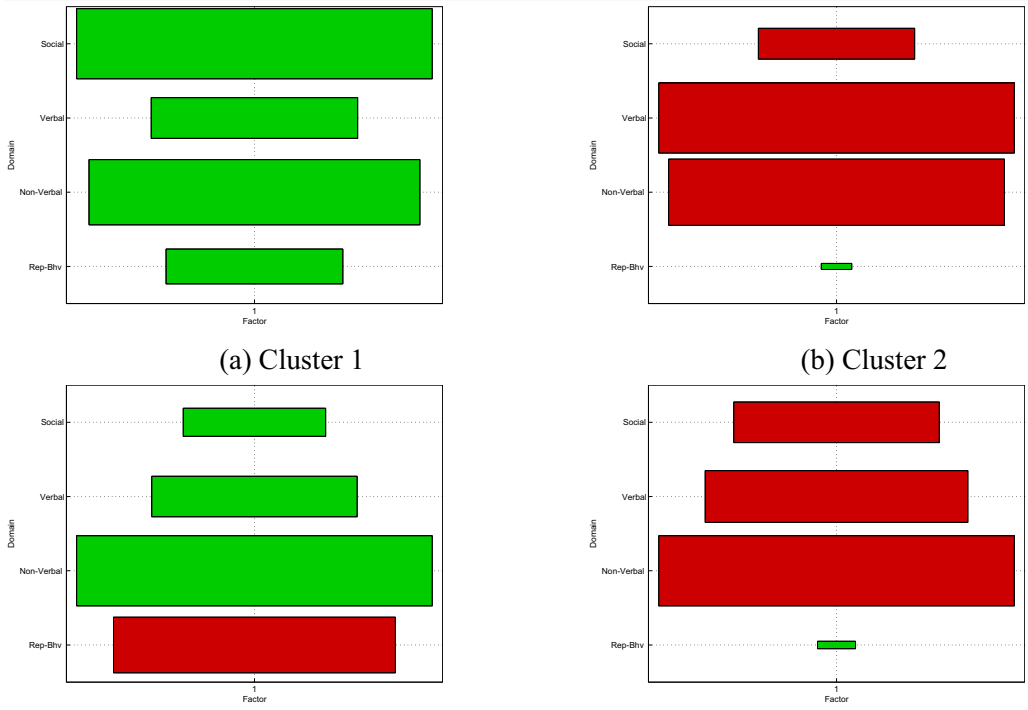
Figure 3.11: Results from the application of the MFA to the ADI-R at domain level, model estimated using BDMCMC. (a): Scores of the subjects in IMGSAC data set, first three domains. Data points are colour coded according to posterior mean cluster assignment with the group of outliers plotted as crosses. The posterior means of the clusters are overlaid as larger markers. (b): Sample path output for K displaying convergence to final value of $K = 5$.

factor is a different combination of features for each group.

Sub-Domain Data

The following results were obtained at the $p = 12$ sub-domain level where the attributes correspond to those specified in Table 3.1. Given the higher dimensionality of the data it was possible to investigate the number of factors per cluster in more detail than at the domain level. It was found that the number of factors q_k varied across the clusters. Figure 3.13 shows the results obtained for K and q_k. The posterior distribution over K peaks at $K = 5$ but also places significant probability on a $K = 3$ and $K = 4$ solution. Further investigation of the $K = 3$ and $K = 4$ solutions showed these configurations to be unstable over repeated estimation and only the more probable and stable $K = 5$ solution was considered.

The samples obtained from the posterior distribution of q_k are plotted in Figure 3.13. The figures clearly shows that for two of the clusters $q_k = 1$ is the dominant solution. However the distribution for the three remaining clusters is much broader and peaks at $q_k = 3$ and $q_k = 5$. We take the median of the samples to obtain integer solutions which



(a) Cluster 1 (b) Cluster 2

(c) Cluster 3 (d) Cluster 4

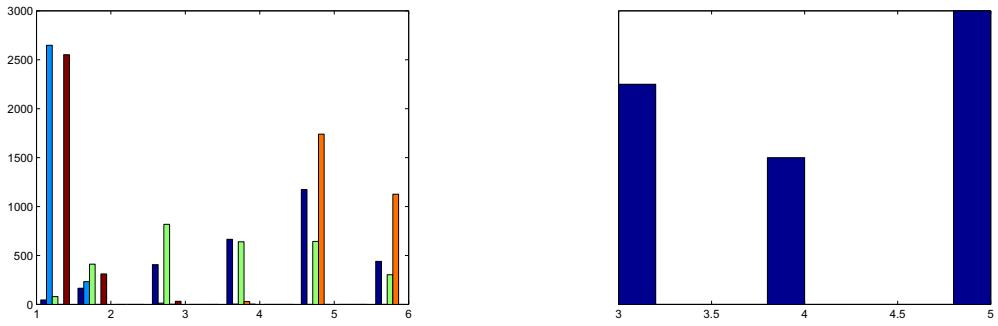
Figure 3.12: Hinton diagrams [61] of the factor loading matrices of the ADI-R clusters at domain scale. A Hinton diagram illustrates the sign and magnitude of the values of the loading matrix. The sign is described by the colour, green positive or red negative and the magnitude of the features is represented by the size of the individual blocks.

results in $\boldsymbol{q} = \{1, 1, 4, 5, 5\}$.

Comparison of the partitioning obtained at the sub-domain level with the domain level $K = 5$ solution produces a Rand index of $.8$ and an adjusted Rand index of $.4$. The latter indicates that there is a strong agreement between the partitions that is greater than what would be expected by chance.

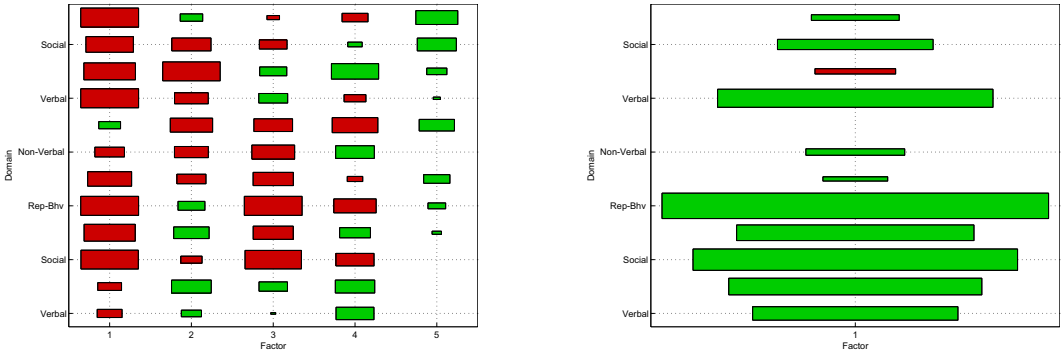
3.5.6 Analysis of ADI-R Using the VBMFA

For completeness the variational method of estimating the MFA, utilised extensively in the empirical study in Section 3.4, was also used to examine the cluster and factor structure of the data. At the domain level the method converged to a $K = 5$ solution and a $K = 4$ solution at the sub-domain level. The Rand index and adjusted Rand index showed



(a) Histogram of $\boldsymbol{q}$ at sub-domain level. (b) Sampled output for K

Figure 3.13: (a): Histogram of the samples obtained from the posterior distribution over q_k at the sub-domain level for $K = 5$. The number of factors per cluster is displayed on the x-axis and ranges from a minimum of one to a maximum of six. The different colours of the bars are to denote different clusters so it can be seen that for two of the clusters the posterior probability for q_k is concentrated at $q_k = 1$ while two other clusters peak at $q_k = 5$. (b): Samples obtained from the posterior distribution over K.



(a) Five factors loading matrix. (b) Single factor loading matrix.

Figure 3.14: Loading matrices of two clusters at the sub-domain level of ADI-R data. Comparison of (a) and (b) reveals that the latent structure varies between different clusters.

strong agreement between both partitions and also between the results obtained using the BDMCMC, adjusted Rand scores all greater than zero. Analysis is restricted to the MCMC results which encapsulate the variational but seem to explore the space better having picked out the non-verbal patients described earlier. Before this the variational results are compared with those obtained using a variational Gaussian Mixture Model in the next section.

3.5.7 Comparison with Mixtures of Gaussians Model

To ascertain whether the clustering obtained using the MFA model on the ADI-R is driven purely by the different means of the clusters or whether $\mathbf{\Lambda}_k$ has a significant impact on the allocation of patients to different clusters, we compare the partitioning obtained with the MFA solved using the variational method to that of a full Gaussian Mixture Model (GMM) estimated by Bayesian variational approximations [14]. The GMM solved in this way requires us to specify the value of K but for the purpose of comparison we use the value obtained from the earlier experiments at both the domain and sub-domain levels. At the domain level we have $K = 5$ and Table 3.4 shows the sizes of the clusters obtained using GMM. It can be seen from the table that no subjects are assigned to the fifth cluster and the model is favouring a $K = 4$ solution using the mixtures of Gaussians. Comparison of this partitioning with the results of the variational MFA partitioning produces an adjusted Rand Index of $.25$. This indicates that there is some agreement in the partitioning obtained using the two methods beyond what would be expected by chance, despite them choosing different values of K.

Cluster	1	2	3	4	5
$K = 5$	66.1%	19.4%	13.8%	1.0%	0.0%

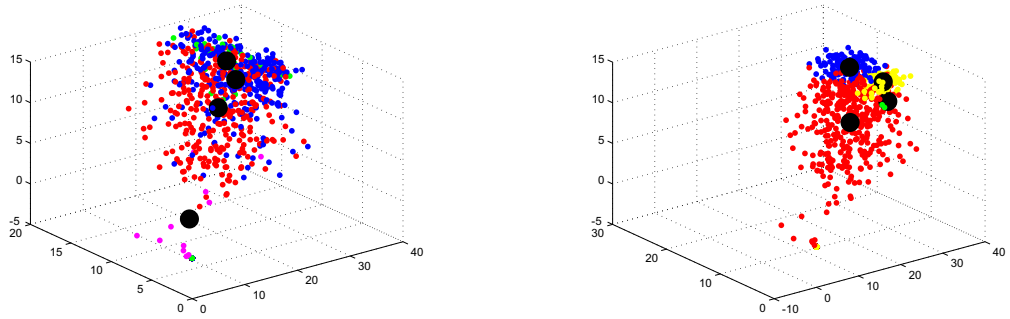
Table 3.4: Relative sizes of the clusters obtained for $K = 5$ using the GMM on the domain level data. No subjects are assigned to the fifth cluster.

Comparison of results at the sub-domain level using $K = 4$ yielded an adjusted Rand Index of $.08$ which indicates that there is almost no agreement between the partitioning using the GMM and that obtained from the MFA. This confirms that the clustering observed is not being solely driven by the mean vectors as the differences in latent structure uncovered by the MFA model impact the clustering. The relative sizes of the clusters at the sub-domain level using the GMM are shown in Table 3.5.

Figure 3.15 shows the partitioning obtained by running the GMM at both the domain and sub-domain levels, but plotted only over the first three dimensions of the domain level. This can be compared visually with the partitioning obtained using the MFA model in

Cluster	1	2	3	4
$K = 4$	48.0%	45.4%	4.6%	2.0%

Table 3.5: Relative sizes of clusters obtained for $K = 4$ using the GMM on the sub-domain data.



(a) GMM domain level partitioning.

(b) GMM sub-domain level partitioning.

Figure 3.15: Partitioning of the patients obtained using the GMM at domain(a) and sub-domain(b) levels. Cluster centers are plotted using large black markers and data points are coloured according to their cluster membership allowing visual comparison with Figure **??** which shows the partitioning obtained using the MFA.

Figure 3.11 The structure obtained in both cases is dominated by one much larger cluster 'super-cluster' which dominates the centre of the data in the GMM results. In the MFA model this cluster is further decomposed into smaller more homogeneous clusters however the overall structure observed is still quite similar even if the composition of the clusters and membership is different, but with the MFA adding more fine detail than the GMM.

3.5.8 Family Structure Analysis

The IMGSAC data is composed of subjects drawn from 299 unique families. This gives us a natural structure or partitioning of the data to compare with that obtained using the MFA. Once again the Rand index can be used to compare the cluster structure with the family structure, by treating each unique family as a cluster. This allows us to investigate whether the partitioning produced by the MFA clusters subjects from the same family together. This may be informative about the clustering mechanism. Upon analysis it was found

that approximately 200 subjects from the total population are partitioned into clusters with a sibling. The Rand indices obtained were .74 and .79, at the domain and sub-domain levels, respectively which seems to imply that the family structure is highly related to the partitioning obtained. However this is due to the fact that we are essentially comparing a $K = 299$ to a $K = 5$ or $K = 4$ solution and the adjusted Rand index value obtained at both levels is approximately zero. This is an effect of the huge disparity between the numbers of clusters. The lack of meaningful agreement between the partitioning obtained and the family structure of the data is still an interesting outcome and will be discussed further in the next section.

3.5.9 Discussion

The results obtained from the analysis of the ADI-R using the MFA are of interest for a number of reasons. Firstly at both levels sub-populations were uncovered within the sample set. This may not be surprising given the heterogeneous nature of ASD, however this specific data set was gathered as part of a genetic study and great effort had been made to obtain homogeneity even using related subjects from the same family. The fact that clusters were still uncovered highlights the difficult problem the heterogeneity of ASD poses. It is reassuring that a group of outliers, that were not a priori known to be present in the data, were picked out and that this cluster has a meaningful interpretation as consisting of patients who need to be assessed in different manner. We now discuss the results at the domain and sub-domain levels in more detail.

Results from the domain level indicate the presence of five clusters within the data. Comparison of the mean of the posterior distribution of the mean for each cluster with the threshold values discussed in Section 3.5.3 reveals that differences in severity exist between each cluster. Subjects in the first three clusters have an average score that is well above the threshold values for diagnosis. However the posterior mean of cluster 4 has much lower deficits in the communication domains and it is possible that within these sub-populations there may be subjects who do not meet the required levels of impairment in all domains to be considered autistic. With this in mind we hypothesise that the single factor could be considered a measure of severity. This ties in with previous findings in [123]

of a continuous severity gradient observed in behavioural data. Under this assumption the different factor loadings obtained for each cluster, defining the latent structure, imply that different domains are more prominent in the level of autism observed for each cluster. Cluster 1 for example is more strongly affected by the social interaction domain that any other clusters.

Interpretation of the results at the sub-domain level is more challenging than at the domain level. Given the broader range of features it becomes more difficult to ascribe meaning to features or judge the level of severity. What is clear from the results at this level though is that clusters are obtained with different intrinsic dimensionalities. This is evidence that the partitioning is not just due to points being located in different regions of space but due to the presence of different latent structures. This is confirmed by the results of the comparison with the GMM and implies that within the population of subjects there are different latent structures being expressed through the manifest behavioural variables.

Comparison of the clustering obtained with the family structure in the data revealed that there was no great tendency for subjects from the same family to be clustered together, beyond what would be expected by chance. This is perhaps unexpected but interesting in itself, the data is behavioural in nature and if we ascribe to the interpretation that the clustering produced at the domain level is illustrative of a severity gradient then what the data is telling us is that within families the severity of symptoms observed can vary widely, to an extent that members of the same family are separated out. This might provide another way of conducting a genetic analysis, rather than group members of the same family together perhaps subjects exhibiting similar behavioural traits should be grouped together to obtain a behaviourally homogeneous sample for genetic analysis.

3.6 Conclusions

This chapter has focused on a Bayesian formulation of the MFA model and two methods for tackling the difficult issue of model selection. In the MFA model this is a two tiered problem with the number of clusters, K, and their intrinsic dimensionalities, q_k, to be determined. We reviewed two methods developed to solve this problem; the variational Bayesian approximation and the stochastic Birth-Death MCMC approach. A simulation

study was conducted to examine the differences in performance obtained from the two methods. This complements a similar study comparing the methods for variable selection in linear regression and extends it to the unsupervised setting for the first time. The results indicated that in situations where the data is sparser the BDMCMC method for estimation provides more consistent results and more accurate detection of clusters within the data. However given sufficient data the variational method can produce matching performance with a fraction of the computational overhead of MCMC. This insight motivates our work in Chapter 5.

Finally we have applied the MFA model estimated using the BDMCMC method to a set of behavioural data from autistic subjects. Analysis of this type data at the domain and sub-domain levels has never previously been conducted using this method. At both levels evidence was found for the existence of sub-populations within the data set which was believed to be homogeneous. At the domain level a key finding was that the clusters formed a severity gradient and the factor loadings of each cluster showed a different relationship between the variables and this factor. Interpretation of results at the sub-domain level was more difficult, however we did find evidence of varying numbers of factors between different clusters, indicating the presence of groups with different latent structures in the data.

The model estimation and selection techniques explored in this chapter will feature again throughout this thesis. The next chapter introduces a latent variable model for analysing network data. The BDMCMC machinery will be used again here allowing us to perform model based clustering of a network in an unsupervised manner, this is a non-trivial problem for which few other methods exist.

Chapter 4

Modelling Network Data using Latent Variables

The previous chapter focussed on the analysis of *multivariate* data using a latent variable model, the MFA. The remaining chapters of this thesis explore the analysis of *relational* data using latent variable methods. More specifically we study *network* data. This type of data has a number of unique characteristics not encountered in multivariate statistics and requiring the development of models and methodology specific to the network representation. The interconnected nature of the data means that the entities or nodes in the network cannot be considered independent. The conditional independence framework of latent variable models provides a convenient way of handling these dependencies, the nodes are independent of each other conditional on an associated latent variable. As well as being convenient the latent variables can also be informative revealing information about the structure of the network for example the number of communities present or allowing the network to be visualised. The Bayesian approach to latent variables utilised thus far will be maintained and we now introduce the particular latent variable models that will be used.

This chapter introduces the Latent Position Model (LPM) [63] and its extension, the Latent Position Cluster Model (LPCM) [58], which incorporates clustering. The model infers a latent space from a network where the positions of the points in latent space are determined by the connections in the network. The model is a latent variable model that generates a multivariate representation of relational data and forms the basis of the remain-

der of the thesis. A number of the ideas introduced in Chapter 3, relating to latent variable models, will feature in this chapter again, for example we extend the LPCM using the birth-death MCMC framework as previously to incorporate inference over the number of clusters, K, in the data. This gives us an unsupervised model based approach for performing community detection in a network which is deployed on both synthetic and real world data.

The chapter is organised as follows: Section 4.1 gives an introduction to the characteristics of network data and some basic models. In Section 4.2 we review the concept of embedding networks into Euclidean space starting with Multi-Dimensional Scaling (MDS) and then introducing the latent variable approach. The model estimation method is then described in Section 4.3. This is an MCMC based approach for which the computational time required scales with $\mathcal{O}(n^2)$. This limits the size of network that the model can feasibly be applied to and in Section 4.4 an approximation to the likelihood, developed in [108], is described that borrows ideas from epidemiology. This case-control likelihood scales linearly with n allowing larger networks to be tackled in reasonable time scales and allowing the model to be extended to incorporate BDMCMC. Section 4.5 describes the application of the Birth-Death MCMC approach to estimating the LPCM for unknown K and we demonstrate this on some synthetic and real data before concluding the chapter.

4.1 Models for Network Data

Networks are becoming an increasingly prevalent method for representing relational data in fields as diverse as social network analysis [78] and neuroscience [124]. This increase in the interest of network structures reflects a shift in approach to the study of complex systems, from concentrating on the elementary system components to encompass knowledge of the ways in which these components interact and the emergent properties of these interactions [124]. Connectivity comes in many forms, for example molecular interactions, metabolic pathways, synaptic connections, emails, social networks, semantic associations, or citation patterns.

The origins of network science can be traced all the way back to Euler in the 18$^{\text{th}}$ century and the *Konigsberg bridge* problem. The problem was to find a path by which a

person could cross each of the seven bridges spanning the city, exactly once and return to the starting point. Euler proved that no such path existed and found a general solution that could be applied to an arbitrary arrangement of bridges and landmasses [41]. More importantly he realised that the problem could be resolved by solely taking into account the relative position of bridges and landmasses and that precise geographical position or physical distance was unimportant. In doing so Euler is generally credited with founding the field which he referred to as the 'geometry of position' (*geometria situs*) and which is now known as graph theory [124]. Although graph theory is considered a branch of pure mathematics it has made a major impact in real-world applications and was used in the derivation of Kirchhoff's laws of voltage and current which laid the foundations of modern circuit theory in electrical engineering [25].

Formally a graph, $G = (V, E)$, is a mathematical structure consisting of a set V of vertices (also referred to as nodes) and a set E of edges denoting the links between vertices. Elements of E are unordered pairs $\{u, v\}$ of distinct vertices $u, v \in V$. In this thesis we focus on simple graphs for which a graph has no edges where both ends connect to a single vertex (called loops) and no pairs of vertices with more than one edge between them (called multi-edges). A graph with either of these properties is called a multi-graph [78].

One of the most elementary representations of a graph is the *adjacency* matrix. The adjacency matrix defines the topology of the graph by representing nodes as matrix rows and columns and representing edges as binary or weighted matrix entries [124]. The ties or edges between nodes may be directed or more simply undirected which results in a symmetric adjacency matrix with $\binom{n}{2}$ possible pairs of connections or dyads [25].

The adjacency matrix allows the derivation of one of the most fundamental graph measures, the *degree*, k. In an undirected graph the degree of a node is the number of edges connected to that node. In directed graphs the indegree and outdegree correspond to the number of incoming and outgoing edges, respectively. The degrees of all nodes together form the degree distribution of the network which shows whether the network contains nodes with approximately equal degrees or whether node degrees vary over a broader range. The degree distribution can be highly informative about the network architecture and we will encounter it again in the following section when discussing some simple network models

and the characteristics of the networks they produce.

Although the origins of network science derive from graph theory, which provides the necessary mathematical foundations and formalism for a coherent study of networks, it has grown and evolved thanks to contributions from a myriad of other fields over the last 70 years. such as sociology [92], statistical physics [38] and probability and statistics [139].

Before 1960 graph theory mainly dealt with the properties of specific individual graphs. In the 1960s, Paul Erdős and Alfred Rényi initiated a systematic study of random graphs [38, 39, 40]. Random graph theory is not the study of individual graphs but the study of a statistical ensemble or probability space of graphs. The ensemble is a class consisting of many different graphs where each graph has a probability attached to it. We will discuss these random graphs as models for networks in more detail in the following section and explain the associated limitations. This development was an important first step in the statistical modelling of networks.

The unique relational nature of network data means that one can encounter certain challenges in statistical analysis that are not present in standard statistics [78]. Statistical modeling is often based on assumptions of independence [119] that samples are i.i.d. observations from a larger population and inference can be made about this population from the sample. This approach does not translate well to the network setting where there are inherent dependencies between nodes and edges due to *reciprocity*, *transitivity*, *homophily* and *assortativity* which are all characteristics observed in networks that we will describe further. As a side note the idea of a network as a set of observations, or subnetwork, from a larger population, analogous to the i.i.d. observations of a population referred to above, is an active area of research with models for networks said to be 'projective' if the same parameters can be used for the full network and for any of its subnetworks [44]. Finding models of this type that can take dyadic dependence into account is an open problem in network science and this problem will not be considered in this thesis.

Transitivity is a key dependency structure seen in networks but most often evident in social network analysis where it is known as the 'friend of a friend is a friend' effect [63]. If there is a tendency toward transitivity the existence of two edges $y_{ij} = y_{jk} = 1$ will lead to an increased probability of there being an edge y_{ik}, the closure of the triangle. Homophily, the tendency of nodes with similar attributes to relate to each other is also common and

leads to a higher probability of links being formed between nodes with similar values on relevant covariates. Reciprocity is the tendency in a directed network for an increase in the probabilty of an edge from node i to j if there is a link from j to i, for example a response email in a communication network. Additionally there is also evidence for correlations between the degrees of nodes a feature known as assortativity [124] which adds to the complexity of the data we are trying to model. There is also evidence for clustering beyond what can be explained by transitivity in many networks such as the formation of cliques in social networks [58].

The above description of the many dependency features observed in networks serves to highlight the complexity of the challenge of modelling such data and developing models that can accurately represent these features. In the next sections we will describe a number of simple network models from the literature which although too simple to account for all these features give further insight to the nature of network data. They will also be useful in later experiments for generating test data. We begin our discussion with the previously mentioned Erdős-Rényi graph.

4.1.1 Simple Network Models

The simplest class of network model is the random network or Erdős-Rényi graph, $G(n, p)$, which for a network of n nodes sets the probability of an edge between each pair of nodes equal to p independently of all other edges [55]. ER graphs can neatly illustrate a key feature of network behaviour known as a 'phase change'. The key to this behaviour is the value $\lambda = pn$. The value $\lambda = 1$ is referred as the percolation threshold in the statistical physics literature and marks the point at which we shift from seeing many small connected components in the form of trees to the emergence of a single 'giant connected component' and the graph becomes fully connected [44]. An example of a fully connected random graph is shown in Figure 4.1 plotted in two dimensions.

In the introduction to this chapter we referred to the degree distribution of a network as being a key characteristic. Erdős-Rényi graphs are characterised as having a Poissonian degree distribution:

$$P(k) = e^{-\lambda}\lambda^k/k!. \tag{4.1}$$

A Poisson degree distribution produces graphs with nodes that have a fairly uniform degree that has a characteristic scale defined by the mean, λ, of the distribution in Equation 4.1. Empirical analysis of the degree distributions of real-world graphs found that this form of degree distribution was a poor match for many real-world examples.

Another simple class of networks is known as the regular lattice graph. In contrast to random graphs, lattice graphs have an ordered pattern of connections between nodes where all nodes have the same number of edges [124]. Examples of regular graphs include the ring, which is shown in Figure 4.2 or grid lattice, where edges link nearby nodes in one or two dimensions.

Random and regular graphs are idealised models that permit some very elegant formal descriptions and analysis. Although these two models are at opposite extremes of the structural spectrum, they both share the essential characteristic that their local structure mirrors (either exactly or statistically) their global structure and hence analysis based on strictly local knowledge is sufficient to capture the statistics of the entire network [139] . However most real world networks are not well described by either random or regular graphs. In the next section a model that combines features of both regular and random graphs is described which has been found to have properties associated with many real world networks.

4.1.2 Watts-Strogatz Model

The modern era of network studies was launched by Duncan Watts and Stephen Strogatz in 1998 [124]. Watts and Strogatz not only devised a deceptively simple network model that explained the origin of the 'small world' phenomenon but also discovered that these patterns are present in a broad range of natural social and technological networks [140]. The model interpolated between a ring lattice and a random network by variation of a single parameter, the probability that an edge of the ring lattice is randomly rewired. If this probability is zero the network is fully regular, if the probability is one the network is fully random. For intermediate settings of the rewiring parameter the graph contains a mixture of regularity and randomness. These random rewirings act as 'shortcuts' in the network connections between nodes and give rise to the 'small world' effect. This is more popularly known as 'six degrees of separation' a trait uncovered by Milgram [92]

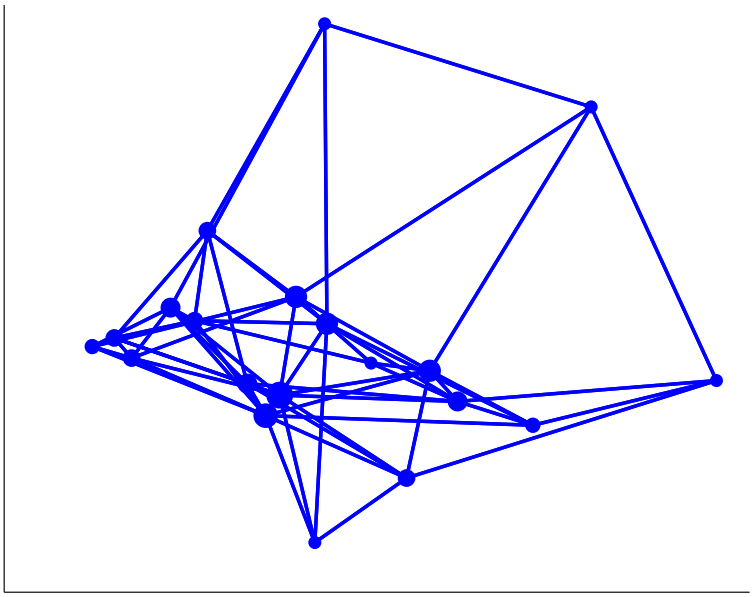
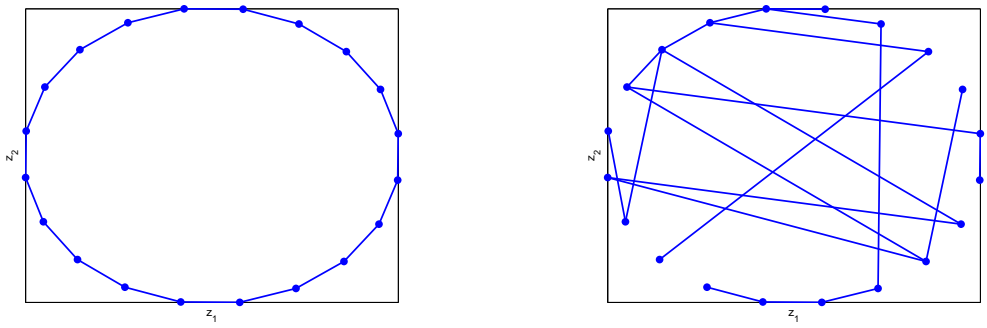


Figure 4.1: Erdős-Rényi network plotted in two dimensions, every node connects randomly with the same probability to create a network with little structure.

in his famous experiment where randomly selected individuals in Kansas and Nebraska were asked to forward a document to target people in Boston. The origin and destination participants were not acquainted and so participants had to forward the document to other acquaintances in a manner that would bring the document closer to its intended target [139]. The average number of intermediary steps was 5.7 which was much smaller than might be expected given the size of the social network, providing empirical evidence of a small world network topology.

4.1.3 Barabási-Albert Model

Traditionally, networks of complex topology were described with the random graph model described in Section 4.1.1. However in the absence of data on large networks the predictions of the ER theory were rarely tested in the real world. When advances in processing power and data acquisition technology allowed for direct measurements of the degree distribution for real networks such as the internet, citation networks, email networks, metabolic



(a) Ring network (b) Small world network

Figure 4.2: Examples of a ring (a) and small world (b) network. The structure of the ring graph is highly regular with connections only existing between neighbouring nodes. In contrast the small world network introduces an element of randomness to the edges.

networks, airline networks, trust networks and neuronal networks, it was found that they did not follow the degree distribution predicted by random graph theory [124]. Rather than a Poissonian degree distribution these networks were found to exhibit a degree distribution that followed a power law:

$$p(k) \sim k^{-\gamma}. \tag{4.2}$$

A power law implies that the probability of finding a node with a degree that is twice as large as an arbitrary number decreases by a constant factor. This result indicates that large networks self organise into a scale free state [25]. Barabasi and Albert [8] demonstrated that these power-law degree distributions could be generated by a *preferential attachment* growth process. This process involves the gradual addition of nodes and the attachment of these nodes to already existing nodes proportional to their degree. A network generated using this process is shown in Figure 4.3 and exhibits the characteristic hubs expected. A common feature of the ER and Watts-Strogatz models is that the probability of finding a highly connected or hub node (large k) decreases exponentially with k; thus vertices with large connectivity are practically absent. In contrast, the power-law tail characterising the degree distribution for the Barabási-Albert model indicate that highly connected nodes have a large chance of occurring and dominating connectivity.

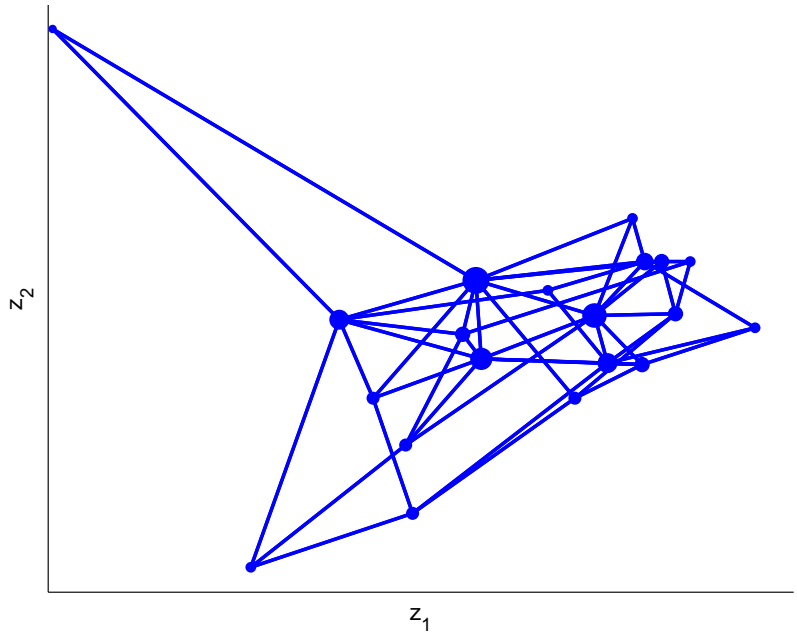


Figure 4.3: Barabási-Albert network generated using a preferential attachment growth method, starting with an initial four nodes, subsequent nodes are added iteratively and link to current nodes dependent upon their degree. Leads to the formation of hubs which we highlight by letting the size of each node be proportional to the degree. Note the last node added in the top left corner connecting to one of the largest hub nodes.

4.1.4 Summary

The simple network models described in this section are parameterised by only one or two parameters which limits the flexibility of the model significantly. While the choice of such a model may be justified in specific cases where the architecture of the network is well understood, for general analysis more complex models are required to accurately model the topology of real networks. As alternatives a number of n-parameter models have emerged in the literature each of which associates a single parameter to every node [105]. The model we have chosen to utilise is the Latent Position Model (LPM) [63] which associates each node with a point in a latent Euclidean space. This has a number of associated advantages which will be discussed in Section 4.2.1 but requires a brief review of the concept of graph embedding which is provided in the next section before the model itself is presented.

4.2 Embedding Networks in Euclidean Space

The concept of distance arises naturally in a network by considering the *path* length. A path is an ordered sequence of unique edges and intermediate nodes that link two nodes. Nodes may be linked directly by an edge, the shortest possible path, or indirectly via sequences of nodes and edges [124]. The *geodesic distance* between two nodes in a network is simply the minimum number of edges one would have to traverse in order to get from one node to the other [95]. All pairwise distances in a network can be represented in the distance matrix [124]. The distance matrix can give insight into the structure of the network. The longest distance or path between two nodes is referred to as the *diameter* of the network and the size of this can be informative as to whether connections are sparse or whether the graph is highly connected with the distance between nodes being small in all cases. In a fully connected undirected graph G the following hold for geodesic distance d between nodes:

$$\begin{aligned} d(u,v) &\geq 0, \\ d(u,v) &= 0, \quad \text{if and only if} \quad u = v \\ d(u,v) &= d(v,u), \\ d(u,v) + d(v,w) &= d(u,w), \end{aligned}$$

for all u and v vertices of G. In an unconnected graph, nodes for which no path exist are said to have infinite distance between them.

Distance naturally implies a set of positions exist. More specifically any given graph can be thought of as a set of points embedded in a Euclidean space, where the Euclidean distance between any two points is just the shortest path length between the corresponding vertices, to within some distortion [140]. The question then arises what is the appropriate dimension of this Euclidean embedding space? Obviously given an $n \times n$ distance matrix it is always possible to embed the network in an n-dimensional space. However, for fixed n, graphs with different topologies will in general have different embedding dimensions. For instance a regular lattice graph such as the ring network shown in Figure 4.2a will have an embedding dimension, $q = 2$, for all n as $n \mapsto \infty$ whereas an ER graph will have an embedding dimension $q = \infty$ as $n \mapsto \infty$. A study of appropriate embedding dimensions

for specific instances of different graphs was conducted in [37] and this question and its relation to the topology of a network continues to be a source of much research [12].

Given a set of distances and an embedding dimension multidimensional scaling (MDS) [28] can be used to determine the optimum positions in that space to minimise a distortion measure. However the distances defined are somewhat arbitrary, for example so far we have only considered the geodesic distance but we could just as easily have used the *detour distance* which utilises the longest path length. Another weakness of the MDS approach is that clusters are better tackled as separate graphs. The limitations of this approach lead us to consider a more principled probabilistic model based approach for the embedding of networks, the Latent Position model.

4.2.1 Latent Position Model

The Latent Position Model (LPM) [63] is a stochastic model of the network in which each node has a latent position in a Euclidean space, and the latent positions are estimated using standard statistical principles; thus no arbitrary choice of dissimilarity is required. The model was originally developed for social network analysis where the probability of an edge between nodes depends on the positions of individuals in an unobserved 'social space'. Various concepts and interpretations of social space have been discussed by [88] and [43] however it is sufficient for our purposes to simply consider this space as a space of unobserved latent characteristics that represent potential transitive tendencies in network relations. Indeed since its conception the model has been further extended and applied to protein-protein interaction data [108] expanding its scope beyond social network analysis.

In Section 4.1 some of the the key features observed in network data, such as transitivity and reciprocation of ties, were discussed as essential elements a suitable network model must capture. The latent position model provides an elegant means of taking transitivity and dependence between dyads into account. The edges in the network are assumed to be conditionally independent given the latent positions, and the probability of a specific edge between two nodes is modeled as some function of their positions such as the distance between the actors in Euclidean latent space [63]. This conditional independence assumption

is defined as:

$$p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{\theta}) = \prod_{i \neq j} (y_{i,j}|\boldsymbol{z}_i, \boldsymbol{z}_j, \beta), \tag{4.3}$$

where $y_{i,j}$ is the presence or absence of a link between nodes i and j, $\boldsymbol{z}_i$ is the position of node i in latent space and β is a parameter to be determined. This should be familiar following our general discussion of latent variable models in Section 2.1.

A convenient parameterisation of $p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{\theta})$ is the logistic regression model in which the probability of a link depends on the Euclidean distance between $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$:

$$\eta_{i,j} = \text{log-odds}(y_{i,j} = 1|\boldsymbol{z}_i, \boldsymbol{z}_j, \beta) = \beta - |\boldsymbol{z}_i - \boldsymbol{z}_j|, \tag{4.4}$$

$$\beta \in \Re, \qquad \boldsymbol{x} \in \Re^q.$$

In Equation 4.4 the distance measure is simply the standard Euclidean norm, in Section 5.3 we will find it convenient to utilise an alternative distance metric, the cosine angle distance. The latent position model is inherently reciprocal and transitive: if $y_{i,j} = y_{j,k} = 1$ then $d_{i,j}$ and $d_{j,k}$ are probably not too large making the events $y_{j,i} = 1$ (reciprocity) and $y_{i,k} = 1$ (transitivity) more probable. The model also provides a framework for the inclusion of covariate information about the nodes to incorporate homophily however we will not require this extension in the thesis where attention is restricted to simple undirected networks with no additional information.

There is often more clustering in a network than can be accounted for purely by transitivity, which can be explained as evidence of cliques or subgroups [58]. A natural extension of this model to incorporate clustering within a network is the Latent Position Cluster Model (LPCM) described in the following section.

4.2.2 Latent Position Cluster Model

The LPCM [58], combines the LPM introduced in the previous section with the mixture model framework described in Section 2.1.4. The model assumes that each node has an unobserved position in a q-dimensional Euclidean latent space as before however to incorporate clustering the latent positions, $\boldsymbol{z}_i$, are assumed to have been drawn from a finite

mixture of G multivariate Gaussian distributions:

$$\boldsymbol{z}_i \sim \sum_{g=1}^{G} \pi_k \mathcal{N}(\boldsymbol{\mu}_g, \sigma_g^2 \boldsymbol{I}_q), \tag{4.5}$$

where π_g are the mixing coefficients and $\boldsymbol{\mu}_g$ is the mean vector of component g. Note that σ_g^2 is a variance parameter specific to each component.

The original paper was received as a read paper in the Royal Statistical Society and we address a number of the comments raised in the accompanying discussions extending the model to allow for automatic variable selection over the value of G.

4.3 Model Estimation Using MCMC

The LPCM is fully specified by Equations 4.3, 4.4 and 4.5 and in contrast to many other network models the log-likelihood of the model is relatively simple and takes the following form:

$$\log p(\boldsymbol{y}|\eta) = \sum_{j \neq i}^{n} \sum_{i \neq j}^{n} \eta_{i,j} y_{i,j} - \log(1 + e^{\eta_{i,j}}), \tag{4.6}$$

where η is defined by Equation 4.4. Calculation of this log likelihood involves a sum over $\frac{1}{2}n(n-1)$ terms for undirected graphs, which increases $\mathcal{O}(n^2)$ for increasing n. The computational burden of calculating the likelihood will make it infeasible for graphs of large n and this is a problem we will address in Section 4.4. However likelihood based estimation methods such as maximum-likelihood and Bayesian inference are feasible [63]. The methods used will be broadly similar to the methods used in Chapter 3 for estimating mixture models, utilising a missing data formulation and MCMC methods. However we must specify the number of clusters, G, a priori. Prior distributions for the model parameters are

specified as in [58]:

$$\begin{aligned}
\beta &\sim \mathcal{N}(\xi, \Psi), \\
\boldsymbol{\pi} &\sim \text{Dir}(\upsilon), \\
\sigma_g^2 &\sim \sigma_0^2 \text{Inv}\chi_2^2, \quad g = 1, \ldots, G \\
\boldsymbol{\mu}_g &\sim \mathcal{N}(\mathbf{0}_p, \omega^2 \boldsymbol{I}_q), \quad g = 1, \ldots, G.
\end{aligned}$$

where $\xi, \Psi, \upsilon, \sigma_0^2$ and ω are hyperparameters to be specified. The MCMC algorithm iterates over the model parameters with the priors given above, the latent positions $\boldsymbol{z}_i$ methods, where possible Gibbs sampling from fully specified posteriors and where this is not possible using Metropolis-Hastings acceptance steps, for the latent positions. We make use of the missing data formulation for mixture models, discussed in Section 2.1.4, and introduce the binary latent allocation vector $\boldsymbol{k}_i$, for each position giving us an $n \times G$ matrix $\boldsymbol{K}$. This is not to be confused with the degree, k, as used earlier. The full conditional posterior distributions are:

$$p(\boldsymbol{z}_i | k_{ig} = 1) \sim \mathcal{N}(\boldsymbol{z}_i; \boldsymbol{\mu}_g, \sigma_g^2 I_p) p(\boldsymbol{y} | \boldsymbol{z}, \beta), \tag{4.7}$$

$$p(\beta) \sim \mathcal{N}(\beta; \xi, \Psi) p(\boldsymbol{y} | \boldsymbol{z}, \beta), \tag{4.8}$$

$$p(\boldsymbol{\pi}) \sim \text{Dir}(m + \upsilon), \tag{4.9}$$

$$p(\boldsymbol{\mu}_g) \sim \mathcal{N}\left(\frac{m_g \bar{z}_g}{m_g + \sigma_g^2/\omega^2}, \frac{\sigma_g^2}{m_g + \sigma_g^2/\omega^2} I \right), \tag{4.10}$$

$$p(\sigma_g^2) \sim (\sigma_0^2 + q\boldsymbol{s}_g^2) \text{Inv}\chi^2_{2+qm_g} \tag{4.11}$$

$$p(k_{ig} = 1) = \frac{\pi_g \mathcal{N}(\boldsymbol{z}_i; \boldsymbol{\mu}_g, \sigma_g^2 I_p)}{\sum_{g=1}^{G} \pi_g \mathcal{N}(\boldsymbol{z}_i; \boldsymbol{\mu}_g, \sigma_g^2 I_p)}, \tag{4.12}$$

where

$$m_g = \sum_{i=1}^{n} [k_{ig} = 1]$$
$$\boldsymbol{s}_g^2 = \frac{1}{q} \sum_{i=1}^{n} (\boldsymbol{z}_i - \boldsymbol{\mu}_g)^T (\boldsymbol{z}_i - \boldsymbol{\mu}_g) I[k_{ig} = 1],$$
$$\bar{\boldsymbol{z}}_g = \frac{1}{m_g} \sum_{i=1}^{n} \boldsymbol{z}_i [k_{ig} = 1].$$

The Metropolis-Hastings algorithm (described in Section 2.3.1) is then used to sample $\boldsymbol{z}_{t+1}$ and β_{t+1} using Equations 4.7 and 4.8. Each position is updated in random order and then parameters $\boldsymbol{k}_i, \boldsymbol{\mu}_g, \sigma_g^2$ and π_g are updated from expressions 4.9 - 4.12.

4.3.1 Implementation Issues

As we are using MCMC methodology a number of issues arise when computing the stochastic average over all iterations. The chief difficulty encountered is that as the likelihood is a function of the latent positions, $\boldsymbol{z}$, only through their distances, it is invariant to reflections, rotations and translations of the latent positions. Thus averaging over the latent positions can produce meaningless results without additional processing. We use the Procrustes transformation [28] which involves translation, rotation, and uniform scaling, to match the posterior draws of latent positions, to a template. This requires the specification of a template position which is chosen as the set of positions which produce the highest value of the likelihood function during the burn-in period. The label switching problem is again encountered and is resolved by the same method used previously in Section 3.2.4.

4.4 Case-control Likelihood

The computational cost of model estimation is of $\mathcal{O}(n^2)$ where n is the number of nodes. This is due to the structure of the likelihood function of the LPCM and makes it infeasible to apply this method to large networks. In order to reduce the computational burden so that it scales linearly with the size of the network we make use of an approximation to the full likelihood known as a case-control likelihood developed and applied to the LPCM in

[108]. In epidemiology case-control studies are widely used to compare a group having an outcome of interest ('case') to a control group with regard to one or more characteristics [17]. Usually the cases are so rare that it is impossible or too expensive to draw a random sample with enough cases. In a case-control study available cases are collected and corresponding controls are sampled from the disease free cohort. Large networks are usually sparse and in this setting the indication of a link (denoted as one in the adjacency matrix) can be considered a case and the absence of a link as controls (indicated by zeros). This analogy suggests an approximation to the log-likelihood function which can be written as follows:

$$l_n \equiv \sum_{j \neq i} \eta_{ij} y_{ij} - \log(1 + e^{\eta_{ij}}), \tag{4.13}$$

$$= \sum_{j \neq i, Y_{ij}=1} \eta_{ij} - \log(1 + e^{\eta_{ij}}) + \sum_{j \neq i, Y_{ij}=0} -\log(1 + e^{\eta_{ij}}), \tag{4.14}$$

$$= l_{i,1} + l_{i,0}. \tag{4.15}$$

The quantity $l_{i,0}$ in Equation 4.15 can be viewed as a population total statistic and can be estimated by a simple random sample of the population i.e.:

$$\hat{l}_{i,0} = \frac{n_{i,0}}{\hat{n}_{i,0}} \sum_{k=1}^{n_{i,0}} -\log(1 + e^{\eta_{ik}}),$$

where $n_{i,0}$ is the total number of zeros in the i^{th} row of the adjacency matrix, $\hat{n}_{i,0}$ is the number of samples selected from the i^{th} row and the sum is over those selected entries. A relatively small $\hat{n}_{i,0}$ can be chosen to get an unbiased estimator of $l_{i,0}$ and thus greatly reduce the amount of computation. This method can be further refined using a stratified sampling approach. The zeros are divided into M strata based on the shortest path length from node i to node j and compute the contribution of the likelihood in the same manner as before. This has the added advantage of increasing the contribution of nodes close together in latent space.

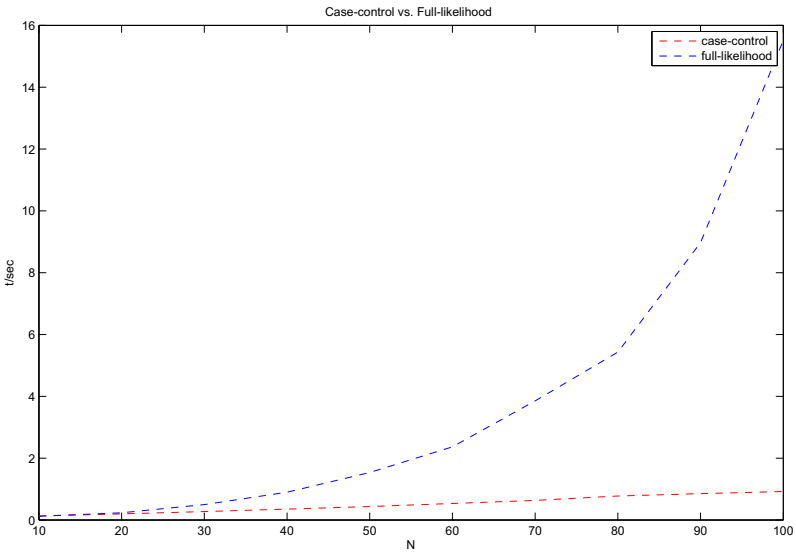


Figure 4.4: Case control likelihood computational times plotted in red against increasing size of network scales linearly in contrast to the full likelihood shown in blue which scales $\mathcal{O}(n^2)$.

4.4.1 Comparison of Case-control and Full

A full comparison of the behaviour of the exact and case-control likelihoods is carried out in [108] where both methods are shown to perform equally well. A brief comparison is provided here to demonstrate how the time required for the same number of iterations scales linearly using case-control and $\mathcal{O}(n^2)$ using the full likelihood. In the experiment we generate a ring network with increasing number of nodes $n = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$ and plot the time taken for a fixed number of iterations. The results are shown in Figure 4.4 and shows that the increasing network size greatly slows down the full likelihood estimation but not in the case-control likelihood. In the next section the increase in computational efficiency is exploited to extend the LPCM to incorporate fully Bayesian model selection of the number of clusters K.

4.5 Automatic Selection of the Number of Clusters

The advantages of the case-control likelihood method for estimation described above are twofold. Firstly now that computational time scales linearly with n it is feasible to apply the model to a much broader range of networks. Secondly this increase in computational speed allows us to consider extensions to the basic model. One issue that needs to be addressed in the model is how to specify the number, K, of mixture distributions used in the latent model. The original paper [58] used the Bayes risk as a method for choosing between different values of K but this requires the model to be estimated for each value of K individually. This is a model selection problem similar to that encountered in Chapter 3 and the methods explored in that chapter are utilised here to perform automatic model selection over K using the birth-death process. This process is described in detail in Chapter 3 but in brief; a Poisson distribution prior is placed on K and the birth-death mechanism is used to draw new clusters from the prior distributions of the parameters using a continuous time Markov chain. An important point to note is that during the model selection process we are not utilising the likelihood function in Equation 4.4 but rather once the positions $\boldsymbol{z}$ have been drawn the mixture distribution likelihood is used to calculate death rates based on these positions. Essentially this is a two stage procedure where first the positions are drawn and then model selection and clustering is performed over these positions.

4.5.1 Example of LPCM with variable K

An example of the birth-death LPCM model's ability to correctly determine the number of clusters present in a network is provided using simulated data obtained by inverting the model. Data is generated composed of three connected clusters of 20 nodes each in two

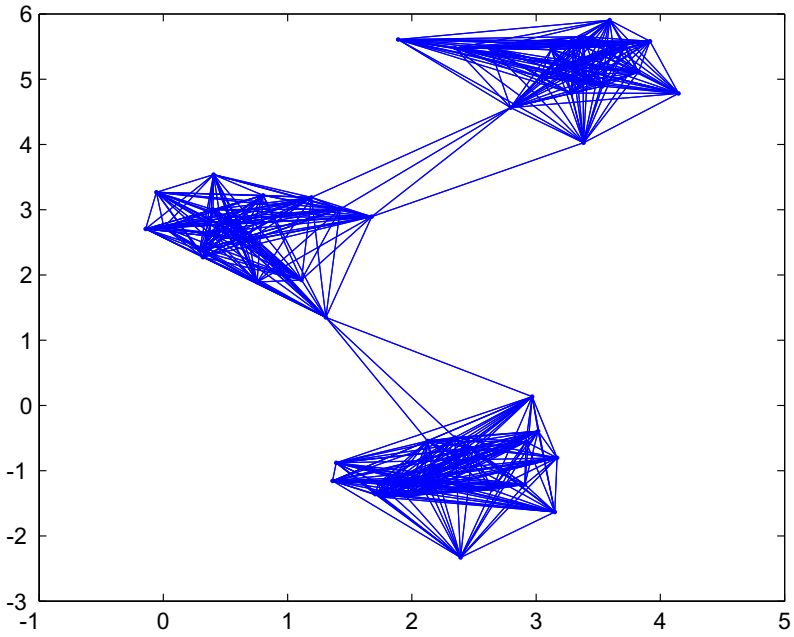
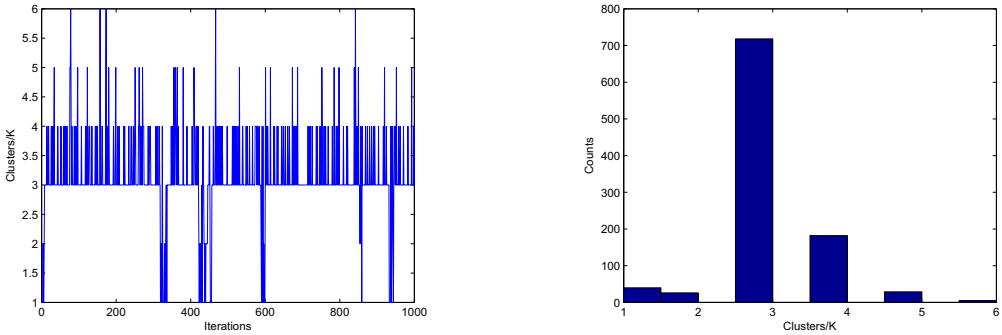


Figure 4.5: Plot of the original positions in space used to generate adjacency matrix and connections subsequently obtained. Data consists of three separate clusters however the network is fully connected.

dimensions. The adjacency matrix $\boldsymbol{Y}$ was simulated in the following manner:

$$
\begin{aligned}
\beta &= 2.0, \\
\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \\ \boldsymbol{\mu}_3 \end{bmatrix} &= \begin{bmatrix} 3.27 & 5.22 \\ 0.63 & 2.67 \\ 2.32 & -1.03 \end{bmatrix}, \\
\sigma^2 &= 0.3, \\
\boldsymbol{z}_i | k_{ig} = 1 &\sim \mathcal{N}_2(\boldsymbol{\mu}_g, \sigma^2 I), \\
y_{ij} &\sim \text{Bernoulli}(\text{logit}^{-1}(\beta - |\boldsymbol{z}_i - \boldsymbol{z}_j|).
\end{aligned}
$$

A plot of the positions and connections generated is shown in Figure 4.5. In Figure 4.6 we show the results obtained using the adjacency matrix as input and running the model for 5000 iterations, after a burn-in period of 5000. As part of the thinning process only every



(a) Sample path K (b) Histogram K

Figure 4.6: Output of the Birth-Death MCMC over K. Figure 4.6a shows the sample path of K for a run of 10,000 samples thinned by taking every 10^{th} sample. The posterior distribution of K is shown in Figure 4.6b and is sharply peaked about the true value $K = 3$.

fifth sample is retained. The posterior distribution over K is shown and is sharply peaked at the correct value of $K = 3$. The model is initialised with $K = 1$ however it is able to determine the number of clusters in the network easily.

4.5.2 VAST Challenge Data

The LPCM with variable K is now used to perform exploratory analysis on a data set of mobile phone records from the IEEE VAST Challenge [56]. The data consists of simulated phone records for an island with 400 unique cell phones. While the data is synthetic it is simulated with sufficient realism that it captures daily cycles in call patterns and can be treated as real. The data consists of ten days worth of cell phone calls and the challenge was to detect a change in the patterns of communication. Previous work on the challenge detected anomalous activity beginning on the eighth day of recording [60]. If the particular change that occurs is sufficient to produce a change in the community structure of the network then by monitoring the value of K in the LPCM we will be able to detect the day the anomalous activity occurs. The data was divided into daily bins embedded with automatic selection of K. The results obtained strongly favoured a $K = 1$ solution on all days and so our community detection method has not allowed us to detect the change. In Section 6.4.2 we will deploy a more sophisticated approach that makes use of the LPCM embedding, to detecting changes in the structure of a network beyond variations in the community

structure. The method makes use of the LPCM as a means of obtaining a multivariate representation of two graphs which can then be compared using CCA as described in detail in Section 6.2.

4.6 Conclusion

This chapter introduced a core element of the thesis, the analysis of network data, maintaining the latent variable approach from the previous chapter. The unique dependency structures inherent in network data were discussed and some simple models introduced that served to highlight some prominent network architectures. These models will be used in later chapters for generating synthetic networks with different structures.

The LPCM was introduced as a method for model based clustering of a network and it will be used extensively in the following two chapters. The model has a number of advantages that make it attractive. The latent positions allow us to take the dependence between dyads naturally into account using the conditional independence framework introduced in Section 2.1. It also provides a model based approach to visualising a network which is an important area of interest in itself.

Crucially the structure of the likelihood of the LPCM makes it amenable to Bayesian analysis. This allows us to take a principled Bayesian approach to model estimation using MCMC which was extended to incorporate a faster approximation to the likelihood developed in [108]. This allowed us to develop perform model selection over the number of clusters, K, using the Birth-Death framework described in Section 3.2.1. The combination of stochastic model selection and this model has not to our knowledge been done before and is a novel contribution. The advantages of Bayesian analysis and model selection have been discussed in the introduction and Section 2.2.1. These are even more pertinent in the network setting. Generally authors in the network literature have often relied upon the extraction of key features of the related graphical network representation, for example, the use of power laws to represent degree distributions or measures of centrality and clustering without any indication that they are either necessary or sufficient as descriptors for the actual network data [44]. The approach favoured here represents a more statistically sound alternative and we will review some methods in Chapter 6 based on the previous approach

for detecting changes in networks and discuss the weaknesses associated.

As well as visualisation of the network the latent positions provide a multivariate representation of the network which allows standard statistical methodology to be deployed on a network, indirectly through this representation. The power of this approach is discussed in [19] as a means of adapting the universe of statistical methodology to the structural pattern recognition field. An unresolved issue here is the embedding dimension of the network, which was briefly discussed in Section 4.2 and made reference to the characteristic embedding dimension of a network. As we have discussed in Chapter 3, from the point of view of factor analysis, the number of factors can have an impact on the cluster structure observed in the network and ideally we would infer the embedding dimension in combination with K as in Section 3.2.3. However this is a very complex task and some of our initial efforts in this area proved unsuccessful, the idea will be addressed again in Section 7.1.2.

In the next chapter we look at an alternative method for model estimation based on variational approximation. Variational theory was introduced in Section 2.3.2 and it has the advantage of generally being a much faster means of estimating model parameters than MCMC methods. The methods for deriving the variational updates can be quite involved and the next chapter is dedicated to deriving these updates for the LPM and LPCM.

Chapter 5

Estimation of the Latent Position Cluster Model by Variational Approximation

A number of complications make the analysis of network data challenging. In the previous chapter we showed how a latent variable model could be deployed to handle the problems posed by the inherent dependencies between nodes in network data. The analysis of network data also suffers from the issue of scale due to the nature of the data which is typically in the form of an $n \times n$ adjacency matrix. As the size of the network increases the associated computational burden rises rapidly. The LPCM, a model based clustering method for embedding graph data in a Euclidean space, was introduced in Chapter 4 and extended to incorporate model selection over the number of clusters. This was based on a MCMC method for model estimation which placed limitations on the sizes of graphs to which we could realistically apply our method to due to the high computational burden. In an effort to overcome these limitations a case-control approximation to the likelihood was introduced which reduced the scaling of the computational burden from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$. However while this extends the range of networks that can be tackled, MCMC based methods are still prohibitively costly. This chapter is concerned with developing an alternative method for model estimation, based on variational Bayesian approximations. In Chapter 3 an empirical comparison between MCMC and variational approximations was performed. This comparison illustrated that given a large dataset the performance obtained from the variational approach was comparable to the MCMC estimation but orders of magnitude

faster.

This insight motivates the following chapter where a method for estimating the LPCM based on variational approximations is developed. An attempt at this has been made recently by other authors in [114], however they were unable to develop fully analytic solutions and resorted to a large number of numerical optimisation routines to obtain model parameters. We tackle the problem from a different perspective utilising an alternative phrasing of the model, as a projection model, as well as a local variational approximation to the logistic function. This approximation has been used extensively in the machine learning literature [14, 94, 69] and has the useful property of having the functional form of a Gaussian. This lends itself to obtaining solutions for the posterior distributions of the model without resorting to numerical approximation.

The chapter is organised as follows. Section 5.1 briefly outlines the 'local' variational approach and then deploys this method to find an approximation to the logistic function. We then attempt to estimate the Latent Position model by variational estimation making use of this approximation to the logistic function. An alternative phrasing of the Latent Position model is presented in Section 5.3 and combining this with the variational approximation from earlier we present a fully variational solution to the Latent Position model in Section 5.4. Building on this in the final section we describe the solution to the more complex LPCM which incorporates clustering in the network.

5.1 Variational Approximation to the Logistic Function

The theory behind variational methods for model estimation in a Bayesian setting is provided in Chapter 2. These methods have been deployed in Chapter 3 as part of the Mixtures of Factor Analysers framework and an empirical comparison with competing MCMC methods was also performed. These can be thought of as 'global' applications of variational methods. In this section a 'local' variational approach will be employed to find an approximation to the logistic function that will allow us to derive fully analytic posteriors for the parameters of the LPCM.

Section 2.3.2 emphasised the importance of the concavity of the log function in developing the lower bound in the global framework. Convexity also plays a central role in

the local variational framework. Consider any continuously differentiable convex function $f(x)$. Convexity of the function guarantees by definition that any tangent line always remains below the function itself. We may thus interpret the collection of all tangent lines as a parameterised family of lower bounds for this convex function [112]. The tangents in this family are naturally parameterised by their locations. From the point of view of approximating the convex non-linear function f it seems natural to use one of the simpler tangent lines as a lower bound. To formulate this a little more precisely let $L(x; x_0)$ be the tangent line at $x = x_0$,

$$L(x; x_0) = f(x_0) + f'(x_0)(x - x_0),$$

then it follows that $f(x) \geq L(x; x_0)$ for all x, x_0 and $f(x_0) = L(x_0; x_0)$. $L(x; x_0)$ is now a variational lower bound of $f(x)$ and x_0 is known as the variational parameter. We have succeeded in approximating the convex function by a simpler, linear function at the expense of introducing an additional parameter x_0 which we must optimise to obtain the tightest bound.

A key component of the Latent Position model and LPCM, see Sections 4.2.1 and 4.2.2, is the logistic function used in Equation 4.4 to estimate the probability of a link or edge based on the distance in latent space between two nodes. The logistic sigmoid arises frequently in probabilistic methods over binary variables, such as presence or absence of an edge, because it is the function that transforms a log odds ratio into a posterior probability. It is defined by

$$\sigma(x) = \frac{1}{1 + e^{-x}}.$$

As it stands the function is neither convex nor concave and so we take transformations of both the input variable and the function itself as in [69] to obtain a convex form. First we

take the log of the logistic function and decompose it so that:

$$\ln \sigma(x) = -\ln (1 + e^{-x}), \tag{5.1}$$
$$= -\ln \{e^{-x/2}(e^{x/2} + e^{-x/2})\}, \tag{5.2}$$
$$= x/2 - \ln(e^{x/2} + e^{-x/2}). \tag{5.3}$$

We now note that the function $f(x) = -\ln (e^{x/2} + e^{-x/2})$ is a convex function of the variable x^2, as can be verified by taking the second derivatives. As discussed above, a tangent surface to a convex funtion is a global lower bound for the function and thus we can bound $f(x)$ globally with a first order Taylor expansion in the variable x^2:

$$\begin{aligned} f(x) &\geq f(\xi) + f'(\xi)(x^2 - \xi^2), \\ &= -\frac{\xi}{2} + \ln \sigma(\xi) + \frac{1}{4\xi}\tanh\left(\frac{\xi}{2}\right)(x^2 - \xi^2), \\ &= -\frac{\xi}{2} + \ln \sigma(\xi) + \lambda(\xi)(x^2 - \xi^2), \end{aligned}$$

where

$$\lambda(\xi) = \frac{1}{4\xi}\tanh\left(\frac{\xi}{2}\right).$$

now substituting into Equation 5.3 and taking exponentials the bound on the logistic function is obtained as:

$$\sigma(x) \geq \sigma(\xi)\exp\{(x - \xi)/2 - \lambda(\xi)(x^2 - \xi^2)\}. \tag{5.4}$$

This bound has the form of the exponential of a quadratic function of x, which will prove useful when we seek Gaussian representations of posterior distributions defined through logistic sigmoid functions.

5.2 Estimation of the Latent Position Model

As mentioned in the introduction to this chapter there have been previous attempts to use variational Bayesian approximations to estimate the latent position cluster model. How-

ever, due to the complexity of the model the update equations for the variational posteriors were analytically intractable requiring numerical optimisation routines to obtain solutions [114]. The performance obtained from this attempt proved to be inferior to that of the original MCMC solution, with the method failing to correctly capture the cluster structure of a benchmark dataset-the Sampson's monk dataset [115]. A variational solution though with performance equal to the original MCMC would be highly desirable due to the superior speed of computation, which would allow application of these methods to ever larger networks. In an effort to obtain a solution we take a different approach to the authors in [114] and make use of the local variational approximation to the logistic function from Section 5.1. The form of this approximation lends itself to computing Gaussian posterior distributions and the hope is this will allow us to compute a fully analytic solution without having to resort to numerical optimisation routines to determine parameters. This will give us a two-fold variational method, taking a 'global' approach to inferring the model while also using a 'local' approach to optimising an approximation to the logistic function.

To begin with we will consider the simpler Latent Position model introduced in Section 4.2.1:

$$p(y_{ij}) = \sigma(\beta - |\boldsymbol{z}_i - \boldsymbol{z}_j|).$$

Here the probability of an edge between two nodes is determined by the Euclidean distance between the respective positions of the nodes in latent space, which are assumed to follow a Gaussian distribution. We replace $|\boldsymbol{z}_i - \boldsymbol{z}_j|$ with u_{ij} which now follows the Rayleigh distribution and utilising the approximation to the logistic function derived previously as Equation 5.4 we have:

$$\begin{aligned}
\ln p(\boldsymbol{y}|\boldsymbol{u},\beta) &\geq \ln h(\beta,\boldsymbol{U},\boldsymbol{\xi}) \\
&= \sum_{i}^{n}\sum_{j}^{n} \ln \sigma(\xi_{ij}) + \left(\frac{(y_{ij}(\beta - u_{ij})) - \xi_{ij}}{2} - \lambda(\xi_{ij})((\beta - u_{ij})^2 - \xi_{ij}^2) \right), \\
&= \sum^{i}\sum^{j} \frac{y_{ij}(\beta - u_{ij})}{2} - \lambda(\xi_{ij})(\beta^2 - 2\beta u_{ij} + u_{ij}^2) \\
&+ \sum^{i}\sum^{j} \left(\ln \sigma(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2 \right).
\end{aligned}$$

In order to determine the variational posterior distribution for $\boldsymbol{u}$ we follow the standard variational approach described in Section 2.3.2 and consider:

$$\ln q^*(\boldsymbol{u}) = \ln h(\beta, \boldsymbol{u}, \xi) + \ln p(\boldsymbol{u}),$$

Substituting in for the appropriate distributions and dropping all terms with no dependence on $\boldsymbol{u}$ we have:

$$\begin{aligned}
\ln q^*(\boldsymbol{u}) &= \sum^{i}\sum^{j} \left(\frac{-y_{ij}u_{ij}}{2} - \lambda(\xi_{ij})(-2\beta u_{ij} + u_{ij}^2) + \ln u_{ij} - \frac{u_{ij}^2}{2\sigma^2} \right), \\
&= \sum^{i}\sum^{j} \left(-\frac{1}{2}\left(y_{ij} + 4\lambda(\xi_{ij})\beta\right) u_{ij} - \{\lambda(\xi_{ij}) + \frac{1}{2\sigma^2}\} u_{ij}^2 + \ln u_{ij} \right).
\end{aligned}$$

The variational posterior distribution for $\boldsymbol{u}$ has a squared, log, and linear term and as there is no distribution of this form a solution is intractable. Our attempts at a variational estimation of the LPM have been thwarted by the Euclidean distance function inherent in the model. However one possible solution is to re-formulate the model as a projection model which will be discussed in the next section.

5.3 Projection Model of Latent Position Model

The Latent Position Model finds an embedding of an adjacency matrix in latent space based on the Euclidean distance between nodes. However alternative distance metrics are equally

valid, for example the Manhattan or Mahalanobis distance. The cosine angle distance, $d = \frac{z_i z_j^T}{|z_j|}$, was proposed as a potentially suitable measure in the original derivations of the LPM [63]. The cosine angle distance is a commonly used measure in content based information retrieval and a comparison of its properties with the Euclidean measure can be found in [107]. In our setting it corresponds to the assumption that nodes i and j are prone to having an edge if the angle between them is small and the model can be reparameterised as:

$$p(\boldsymbol{y}|\boldsymbol{z}, \beta) = \prod_i^n \prod_j^n \sigma(y_{ij}(\beta + \frac{\boldsymbol{z}_i \boldsymbol{z}_j^T}{|\boldsymbol{z}_j|})).$$

Note that in particular we no longer have the problematic $|\boldsymbol{z}_i - \boldsymbol{z}_j|$ term. Instead we now have the unit vector $\frac{z_j}{|z_j|}$ which we replace with the unit vector $\boldsymbol{u}_j$ as in Section 5.2 but no longer referring to $|\boldsymbol{z}_i - \boldsymbol{z}_j|$. The unit vector $\boldsymbol{u}_j$ lies on the unit circle so it also cannot be said to follow a Gaussian distribution. Instead we will make use of a distribution commonly used in the study of directional data, the von Mises distribution.

5.3.1 Von Mises Distribution

Mapping data to the unit circle allows it to be described in polar form simply by the angle θ, measured in radians, as $\boldsymbol{z}_1 = (\cos\theta, \sin\theta)$. The angle θ is a periodic variable and the use of standard statistical methods to study its distribution is inappropriate. To illustrate the potential problems encountered with periodic variables we briefly switch to measuring angles in degrees, if the mean of angles $\theta_1 = 1^o$ and $\theta_2 = 359^o$ is modeled as a standard univariate Gaussian with origin at 0^o the value would be 180^o with standard deviation 179^o which is clearly counterintuitive [84]. Instead we require specific distributions for periodic variables $p(\theta)$ and these must satisfy the three conditions [14]:

$$\begin{aligned} p(\theta) &\geq 0, \\ \int_0^{2\pi} p(\theta) d\theta &= 1, \\ p(\theta + 2\pi) &= p(\theta). \end{aligned}$$

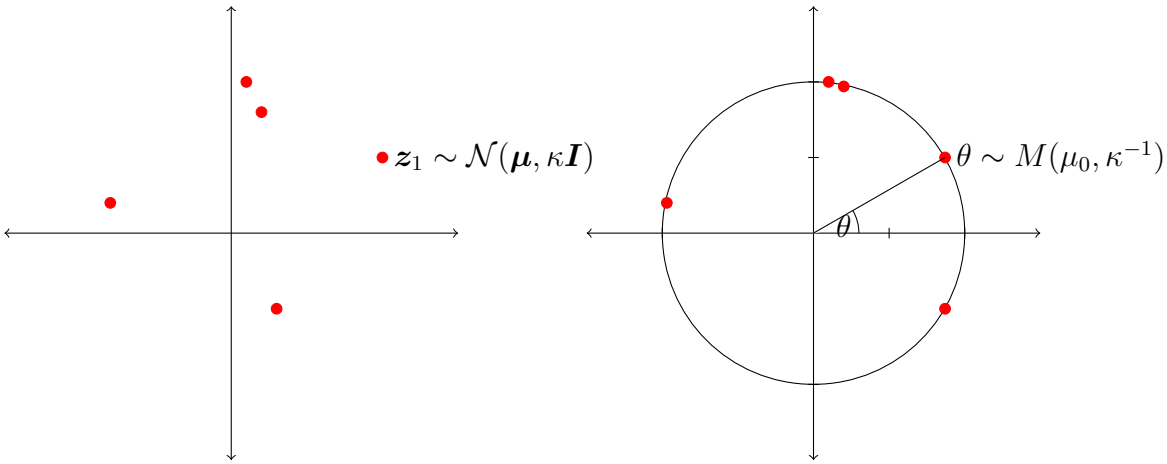
The most widely used distribution on the circle is the von Mises distribution [47]. It has a number of features which make it appealing for our uses, particularly it's close relationship with the Gaussian distribution. The von Mises distribution $M(\mu, \kappa)$ has probability density function:

$$f(\theta; \mu, \kappa) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa \cos(\theta - \mu_0)},$$

where I_0 denotes the modified Bessel function of the first kind and order 0 which can be defined by:

$$I_0(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} e^{\kappa \cos\theta} d\theta.$$

The parameter μ_0 is the mean direction and the parameter κ is known as the concentration parameter [84], not to be confused with the sufficient statistic, variance. The von Mises



(a) Data in plane (b) Data mapped to unit circle

Figure 5.1: Relationship between Gaussian and von Mises distribution. (a): the data is shown in a plane and can be described by a bivariate Gaussian distribution. (b): data mapped to the unit circle and they are now described using a von Mises distribution on the angle θ.

distribution is related to the bivariate Gaussian distribution as follows. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be independent Gaussian variables with means $(\cos \mu_0, \sin \mu_0)$ and equal variances $1/\kappa$. This construction ensures that the mean lies on the unit circle. The p.d.f. of the polar variables

(r, θ) is proportional to:

$$r \exp\left(-\frac{\kappa}{2}\{r^2 - 2r\cos(\theta - \mu_0)\}\right).$$

Since the range of r does not depend on θ the conditional distribution for $r = 1$ is the von Mises distribution $M(\mu_0, \kappa)$. The above relations of the von Mises distribution with the Gaussian distribution clearly indicate that μ_0 behaves like the mean while $1/\kappa$ influences the von Mises distribution in the same way that σ^2 influences the Gaussian distribution. For $\kappa = 0$ the distribution is uniform while for large κ the distribution is clustered around the direction μ_0 [14].

As part of the variational estimation we will be inferring distributions over the positions of the individual nodes in latent space. These calculations require the evaluation of expectations with respect to these positions, $\boldsymbol{z}$, projected on to the unit circle, $\boldsymbol{u}$. The relationship between the von Mises distribution and the Gaussian distribution described above allows us to estimate the parameters of the appropriate von Mises distribution on the circle from the distribution of the latent positions. This is illustrated in Figure 5.1 where we show data in the plane which is then mapped to the unit circle. If the latent position has an associated Gaussian distribution, $\boldsymbol{z}_1 \sim \mathcal{N}(\boldsymbol{\mu}, \kappa \boldsymbol{I})$, then the angle of the transformed variable has an associated von Mises distribution $\theta \sim M(\mu_0, \kappa^{-1})$.

The relationship between the Gaussian distribution and the von Mises distribution allows us to infer the parameters of the distribution of the transformed positions, $\boldsymbol{u}$, from the original positions, $\boldsymbol{z}$. These are required to evaluate the expectations in the variational updates equations. A drawback of using this distribution however is that we are now restricted to a 2-D latent space as well as using isotropic covariances matrices to preserve this relationship. The Latent Position model was originally derived using isotropic covariance matrices so this is not that much of a limitation. However the restriction of the embedding dimension is undesirable. The Fisher-Bingham distribution [74]is a distribution that generalises the von Mises distribution to spaces of higher dimension and its use could potentially allow for more flexibility in the choice of embedding dimension in the algorithm. However, the increased complexity that this involves means that for the time being we will work with the 2-D embedding space. The dimension of the embedding space is an important issue

which was touched upon in Section 4.2 and will be discussed again in Section 7.1.

In the following section we perform a simulation with synthetic data generated from the von Mises distribution and a Gaussian distribution mapped to the unit circle to further demonstrate the transformation property that makes it so useful in our application.

5.3.2 Empirical Demonstration of Relationship Between the Gaussian and von Mises distributions

In this section we conduct a simulation with synthetic data to demonstrate the relationship described previously between the von Mises and bivariate Gaussian distribution. The experiment is described in Algorithm 2. The results from the experiment are displayed in

Algorithm 2 Comparison of Gaussian and von Mises distributions

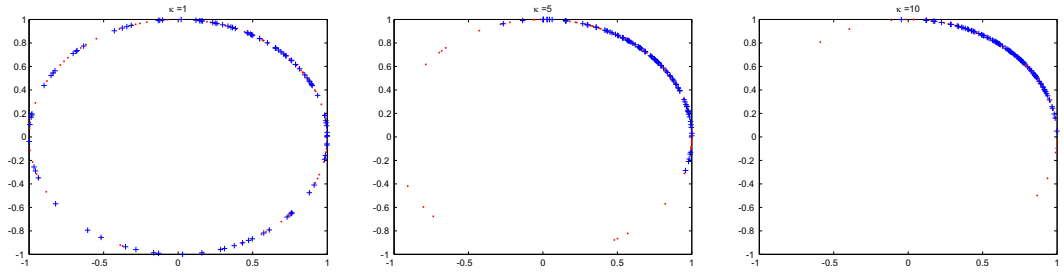
for $\kappa = 1:9$ and $n = 100$ **do**
Draw n samples $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}, \kappa \boldsymbol{I})$
Map these samples to the unit circle $\boldsymbol{z} \mapsto \boldsymbol{u}$, $\boldsymbol{u} = \frac{z}{||z||}$.
Calculate μ_0 and κ for the von Mises distribution that match $\mathcal{N}(\mu, \kappa \boldsymbol{I})$.
Draw n samples $\theta \sim M(\mu_0, \kappa^{-1})$
Plot transformed $\boldsymbol{z}$ against von Mises samples θ
end for

Figure 5.2. The results show that samples drawn from the Gaussian distribution and then mapped to the unit circle are closely matched by those drawn from the von Mises distributions whose parameters are chosen to match that of the original Gaussian distribution but on the circle. As we change the variance of the Gaussian distribution the resulting change in the value of κ produces the same effect observed in the mapped samples.

5.4 Estimation by Variational Approximation

We now return to determining the variational update equations for the Latent Position model utilising the new tools we have just described, the von Mises distribution and projection form of the model. As before the approximation to the logistic is:

$$\sigma(\boldsymbol{z}) \geq \sigma(\xi) \text{exp}\left(\frac{\boldsymbol{z} - \xi}{2} - \lambda(\xi)(\boldsymbol{z}^2 - \xi^2)\right),$$



(a) $\kappa = 1$ (b) $\kappa = 5$ (c) $\kappa = 10$

Figure 5.2: Plots of mapped samples from the Gaussian distribution in red with the samples drawn from the corresponding von Mises distribution overlaid in blue. As the variance of the Gaussian distribution is reduced $\sigma^2 = 1, \frac{1}{5}, \frac{1}{10}$ the corresponding value of the concentration parameter κ increases and the behaviour of the samples from the von Mises distribution closely follows that of the mapped samples.

which admits a conjugate prior in the exponential family. According to the latent position model:

$$p(\boldsymbol{y}|\boldsymbol{z}, \beta) = \prod_{i}^{n} \prod_{j}^{n} \sigma(y_{ij}(\beta - |\boldsymbol{z}_i - \boldsymbol{z}_j|)).$$

Replacing $|\boldsymbol{z}_i - \boldsymbol{z}_j|$ with $\beta + \frac{z_i z_j^T}{|z_j|}$ the Latent Position model now takes the form:

$$p(\boldsymbol{y}|\boldsymbol{z}, \beta) = \prod_{i}^{n} \prod_{j}^{n} \sigma(y_{ij}(\beta + \frac{\boldsymbol{z}_i \boldsymbol{z}_j^T}{|\boldsymbol{z}_j|}))$$

We now substitute the variable $\boldsymbol{u}_j$ for $\frac{z_j}{|z_j|}$ which follows the von Mises distribution, utilising the lower bound for the logistic function, Equation 5.4, the log marginal likelihood of the data is lower bounded by:

$$\ln p(\boldsymbol{y}|\boldsymbol{u},\beta) \geq \ln h(\boldsymbol{U},\beta,\boldsymbol{\xi}) \tag{5.5}$$

$$= \sum_{i}^{n}\sum_{j}^{n} \ln \sigma(\xi_{ij}) + \left(\frac{(y_{ij}(\beta + \boldsymbol{z}_i\boldsymbol{u}_j^T)) - \xi_{ij}}{2} - \lambda(\xi_{ij})((\beta + \boldsymbol{z}_i\boldsymbol{u}_j^T)^2 - \xi_{ij}^2) \right), \tag{5.6}$$

$$\begin{aligned} &= \sum^{i}\sum^{j} \frac{y_{ij}(\beta + \boldsymbol{z}_i\boldsymbol{u}_j^T)}{2} - \lambda(\xi_{ij})(\beta^2 + 2\beta\boldsymbol{z}_i\boldsymbol{u}_j^T + \boldsymbol{z}_i\boldsymbol{u}_j^T\boldsymbol{u}_j\boldsymbol{z}_i^T), \\ &+ \sum^{i}\sum^{j} \left(\ln \sigma(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2 \right). \end{aligned} \tag{5.7}$$

Now to update the model we consider the posterior distribution of the individual $\boldsymbol{z}_i$ assuming independence. The latent positions are initialised, and the statistical properties of the individual $\boldsymbol{u}_j$ inferred and the parameters of the distribution of $\boldsymbol{u}$. The variational posteriors are evaluated by standard variational methods for factorised distributions, described in Section 2.3.2:

$$\begin{aligned} \ln q^*(\boldsymbol{z}_i) &= \mathbb{E}_{u,\beta}\left[\ln h(\boldsymbol{z}_i,\boldsymbol{u},\beta,\xi) + \ln p(\boldsymbol{z})\right], \\ &= \mathbb{E}_{u,\beta}\Bigg[\sum^{j} \frac{y_{ij}(\beta + \boldsymbol{z}_i\boldsymbol{u}_j^T)}{2} - \lambda(\xi_{ij})(\beta^2 + 2\beta\boldsymbol{z}_i\boldsymbol{u}_j^T + \boldsymbol{z}_i\boldsymbol{u}_j^T\boldsymbol{u}_j\boldsymbol{z}_i^T) \\ &+ \sum^{j} \left(\ln \sigma(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2 \right) - \frac{1}{2}\boldsymbol{z}_i I \boldsymbol{z}_i^T \Bigg]. \end{aligned}$$

Now dropping terms with no dependence on $\boldsymbol{z}_i$ and rearranging:

$$\begin{aligned} &= \mathbb{E}_{u,\beta}\left[\sum^{j} \frac{y_{ij}(\boldsymbol{z}_i\boldsymbol{u}_j^T)}{2} - \lambda(\xi_{ij})(2\beta\boldsymbol{z}_i\boldsymbol{u}_j^T + \boldsymbol{z}_i\boldsymbol{u}_j^T\boldsymbol{u}_j\boldsymbol{z}_i^T) - \frac{1}{2}\boldsymbol{z}_i I \boldsymbol{z}_i^T \right], \\ &= \mathbb{E}_{u,\beta}\left[z_i \left(\sum^{j} \frac{y_{ij}}{2}\boldsymbol{u}_j - 2\beta\lambda(\xi_{ij})\boldsymbol{u}_j \right) - \frac{1}{2}\boldsymbol{z}_i\Big(\sum^{j} I + 2\lambda(\xi_{ij})\boldsymbol{u}_j^T\boldsymbol{u}_j \Big)\boldsymbol{z}_i^T \right]. \end{aligned}$$

Taking expectations with respect to β:

$$= z_i \left(\mathbb{E}_u \left[\sum^{j} \frac{y_{ij}}{2} \boldsymbol{u}_j - 2\mathbb{E}_\beta[\beta]\lambda(\xi_{ij})\boldsymbol{u}_j \right] \right) - \frac{1}{2}\boldsymbol{z}_i \Big(\sum^{j} I + 2\lambda(\xi_{ij})\mathbb{E}_u[\boldsymbol{u}_j^T \boldsymbol{u}_j] \Big) \boldsymbol{z}_i^T,$$

which is a Gaussian distribution with parameters:

$$\begin{aligned} \boldsymbol{\Sigma}_{z_i}^{-1} &= \Big(\sum^{j} I + 2\lambda(\xi_{ij})\mathbb{E}_u[\boldsymbol{u}_j^T \boldsymbol{u}_j] \Big), \\ \boldsymbol{\mu}_{z_i} &= \boldsymbol{\Sigma}_{z_i} \left(\mathbb{E}_U \left[\sum^{j} \frac{y_{ij}}{2} u_j - 2\mathbb{E}_\beta[\beta]\lambda(\xi_{ij})\boldsymbol{u}_j \right] \right). \end{aligned}$$

The variational posterior requires the evaluation of $\mathbb{E}[\beta]$ and as expected the variational update equations are coupled.

5.4.1 Posterior Update Equations for β

We place a zero mean unit variance Gaussian prior on β and utilise the same approximation, the optimal solution for the variational posterior distribution is:

$$\begin{aligned} \ln q^*(\beta) &= \mathbb{E}_{u,z}[\ln h(\boldsymbol{z}, \boldsymbol{u}, \beta, \boldsymbol{\xi}) + \ln p(\beta)], \\ &= \mathbb{E}_{u,z}\Big[\sum^{i} \sum^{j} \frac{y_{ij}(\beta + \boldsymbol{z}_i \boldsymbol{u}_j^T)}{2} - \lambda(\xi_{ij})(\beta^2 + 2\beta \boldsymbol{z}_i \boldsymbol{u}_j^T + \boldsymbol{z}_j \boldsymbol{u}_i^T \boldsymbol{u}_i \boldsymbol{z}_j^T) \\ &+ \sum^{i} \sum^{j} \left(\ln \sigma(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2 \right) - \frac{1}{2}\beta \boldsymbol{I} \beta \Big]. \end{aligned}$$

Dropping terms with no dependence on β we have:

$$\begin{aligned} \ln q^*(\beta) &= \mathbb{E}_{u,z}\Big[\sum^{i} \sum^{j} \frac{y_{ij}}{2}\beta - \lambda(\xi_{ij})(\beta^2 + 2\beta \boldsymbol{z}_i \boldsymbol{u}_j^T) - \frac{1}{2}\beta \boldsymbol{I} \beta \Big], \\ &= \beta \left(\frac{n_{\text{edges}}}{2} - \sum^{i} \sum^{j} 2\lambda(\xi_{ij})\mathbb{E}_z[\boldsymbol{z}_i]\mathbb{E}_u[\boldsymbol{u}_j^T] \right) \\ &- \frac{1}{2}\beta^T \left(\boldsymbol{I} + 2 \sum^{i} \sum^{j} \lambda(\xi_{ij}) \right) \beta. \end{aligned}$$

which we recognise as a Gaussian distribution with parameters:

$$\boldsymbol{\Sigma}^{-1} = \left(\boldsymbol{I} + 2 \sum^{i} \sum^{j} \lambda(\xi_{ij}) \right), \tag{5.8}$$

$$\boldsymbol{\mu}_{\beta} = \boldsymbol{\Sigma} \left(\frac{n_{\text{edges}}}{2} - \sum^{i} \sum^{j} 2\lambda(\xi_{ij}) \mathbb{E}_z[z_i] \mathbb{E}_u[\boldsymbol{u}_j^T] \right), \tag{5.9}$$

where n_{edges} is the number of edges in the network.

5.4.2 Variational Parameters

To estimate the values of the variational parameters we need only consider the factors that depend on ξ, the approximation to the log-likelihood:

$$\ln h(\xi, \beta, \boldsymbol{z}) = \sum^{i} \sum^{j} \ln \sigma(\xi_{ij}) - \xi_{i,j}/2 - \lambda(\xi_{ij}) \left((\beta + \boldsymbol{z}_i \boldsymbol{u}_j^T)^2 - \xi_{ij}^2 \right). \tag{5.10}$$

Taking derivatives with respect to ξ and setting to zero as in [14] we obtain:

$$0 = \lambda'(\xi_{ij}) \left((\beta + \boldsymbol{z}_i \boldsymbol{u}_j^T)^2 - \xi_{ij}^2 \right),$$

we now note that $\lambda'(\xi)$ is a monotonic function of ξ for $\xi \geq 0$, and that we can restrict attention to nonnegative values of ξ without loss of generality due to the symmetry of the bound around $\xi = 0$. Thus $\lambda'(\xi) \neq 0$ and hence we obtain the following re-estimation equation:

$$\xi_{ij}^2 = \mathbb{E}_{\beta,z,u}(\beta^2 + 2\beta \boldsymbol{z}_i \boldsymbol{u}_j^T + \boldsymbol{z}_i \boldsymbol{u}_j^T \boldsymbol{u}_j \boldsymbol{z}_i^T), \tag{5.11}$$

$$= \mu_{\beta}^2 + \text{Tr}(\boldsymbol{\Sigma}_{\beta}) + 2\mu_{\beta} \boldsymbol{z}_i \boldsymbol{u}_j^T + \text{Tr}(\boldsymbol{S}_i \boldsymbol{S}_j), \tag{5.12}$$

where

$$\boldsymbol{S}_i = \boldsymbol{z}_i^T \boldsymbol{z}_i + \boldsymbol{\Sigma}_Z, \tag{5.13}$$

$$\boldsymbol{S}_j = \boldsymbol{u}_j^T \boldsymbol{u}_j + \kappa^{-1}. \tag{5.14}$$

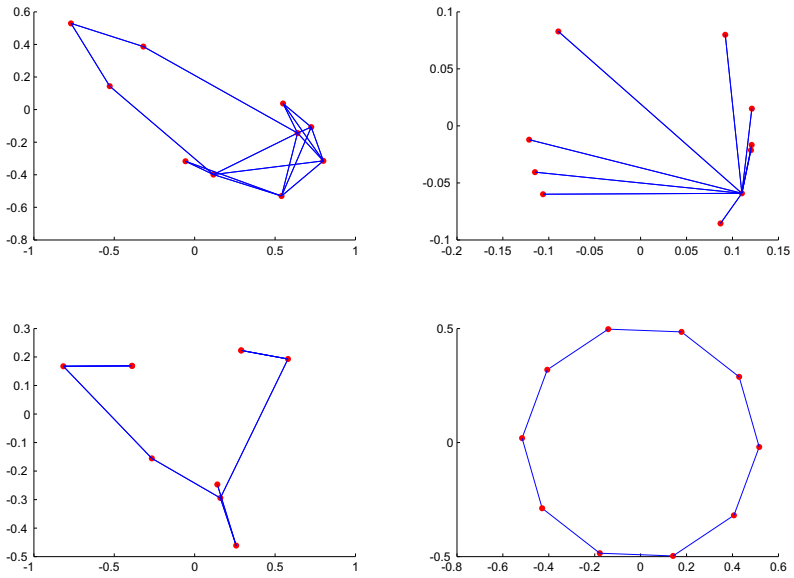


Figure 5.3: Results from the embedding of four different adjacency matrices, ER, star, tree and ring graphs. $n = 10$ in all cases.

5.4.3 Results

The equations derived in the previous sections allow us to use the variational method to fit the Latent Position model for a given adjacency matrix. This is done by repeatedly cycling through the distributions which are dependent upon each other in the same manner as an EM algorithm. To verify the algorithm is working correctly we test it on four standard networks previously encountered in Chapter 4, the Erdos-Renyi network, star network, tree network and ring network. The results obtained from deploying the variational Bayesian latent position model on instances of these networks with $n = 10$ nodes are shown in Figure 5.3 and match what would be expected given the form of the networks. The variational model estimation method has been verified to function as expected on a set of small networks. However, the motivation for the method's development was the belief it would be better suited to large scale networks than the MCMC equivalent. We now demonstrate estimation of larger scale versions of the same four networks with $n = 500$ using the variational approach. This is a challenging sized network with such methodology but still within the reach of the case-control version of MCMC, see Section 4.4, however the experiment

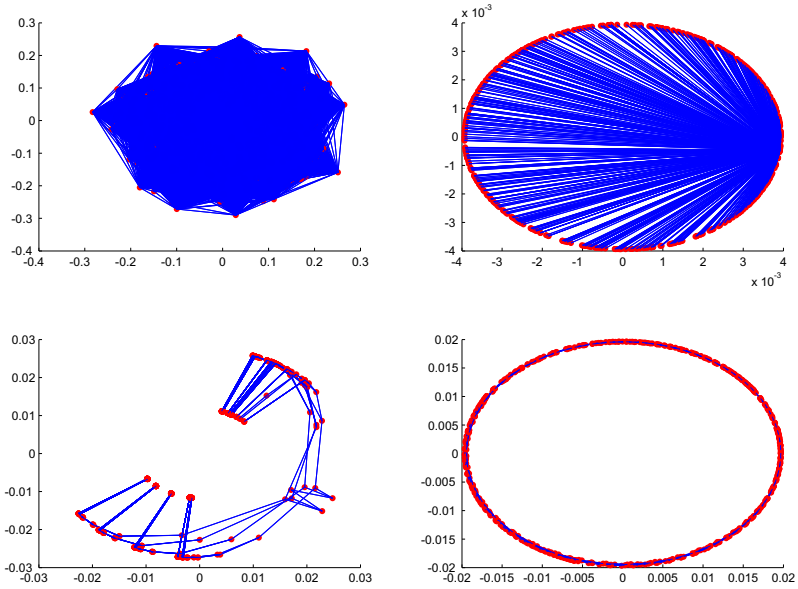


Figure 5.4: Results from the embedding of four different adjacency matrices, ER, star, tree and ring graphs. $n = 500$ in all cases.

involving all four networks terminates considerably faster than the computations required for a single MCMC estimation. The results are shown in Figure 5.4. Although there are difficulties in visualising such large networks the general structure observed matches what we would expect.

5.5 Variational Estimation of the LPCM

We now extend the approach demonstrated in the previous sections to the more complex LPCM. The LPCM, introduced in Section 4.2.2, is an extension of the Latent Position model that incorporates clustering in the data. To represent the clustering the latent positions, $\boldsymbol{z}$, are modeled as coming from a mixture of K multivariate Gaussian distributions:

$$\boldsymbol{z}_i \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\mu}_k, \sigma_k^2 I).$$

For convenience we will work with the precision, $\mathbf{\Lambda}_k$ rather than the variance. The parameters $\beta, \boldsymbol{\pi}, \lambda$ and $\boldsymbol{\mu}$ are also given prior distributions and the full set of priors is:

$$\boldsymbol{z}_i \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\mu}_k, \mathbf{\Lambda}_k^{-1}), \tag{5.15}$$

$$p(S|\boldsymbol{\pi}) = \prod^{N}\prod^{K} \pi_k^{s_{nk}}, \tag{5.16}$$

$$p(\boldsymbol{z}|\boldsymbol{S}, \boldsymbol{\mu}, \sigma^2) = \prod^{N}\prod^{K} \mathcal{N}(\boldsymbol{z}_n|\boldsymbol{\mu}_k, \Lambda_k^{-1})^{s_{ng}}, \tag{5.17}$$

$$p(\boldsymbol{\pi}) \sim \text{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha}) = C(\alpha)\prod^{K} \pi_k^{\alpha-1}, \tag{5.18}$$

$$p(\boldsymbol{\mu}, \mathbf{\Lambda}) \sim \mathcal{N}(\boldsymbol{m_0}, (\omega_0 \mathbf{\Lambda}_k)^{-1})\mathcal{W}(\mathbf{\Lambda}_k|W_0, \nu_0). \tag{5.19}$$

The relationships between the parameters of the model are best expressed in the form of a Directed Acyclic Graph shown in Figure 5.5 This shows the structure of the joint dis-

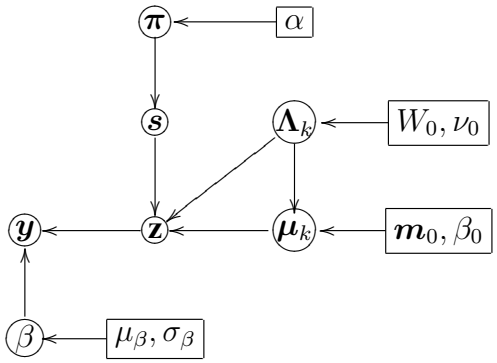


Figure 5.5: Directed acyclic graph for the latent position cluster model where Z represents the latent positions and Y the data in the form of the adjacency matrix of connections.

tribution of all of the random variables in the model and allows us to write the marginal likelihood as:

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{z}, \beta)p(\boldsymbol{z}|\boldsymbol{s}, \boldsymbol{\mu}, \mathbf{\Lambda}^{-1})p(\boldsymbol{s}|\boldsymbol{\pi})p(\boldsymbol{\pi})p(\boldsymbol{\mu}, \mathbf{\Lambda})p(\beta|\mu_\beta, \sigma_\beta^2). \tag{5.20}$$

The optimal variational distributions can be found using the standard variational methodology of taking expectations with respect to the other parameters:

$$\ln q^*(\boldsymbol{z}) = \mathbb{E}_{\beta}[\ln p(\boldsymbol{y}|\boldsymbol{z},\beta)] + \mathbb{E}_{s,\mu,\Lambda}\ln\left[p(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{\mu},\boldsymbol{\Lambda}\right], \tag{5.21}$$

$$\ln q^*(S) = \mathbb{E}_{\pi}[\ln p(\boldsymbol{z}|\boldsymbol{\pi})] + \mathbb{E}_{\mu,\Lambda^{-1}}[\ln p(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{\mu},\boldsymbol{\Lambda}])\text{,} \tag{5.22}$$

$$\ln q^*(\boldsymbol{\pi}) = \ln p(\boldsymbol{\pi}) + \mathbb{E}_{s}[\ln p(\boldsymbol{s}|\boldsymbol{\pi})], \tag{5.23}$$

$$\ln q^*(\boldsymbol{\mu},\boldsymbol{\Lambda}) = \ln p(\boldsymbol{\mu},\boldsymbol{\Lambda}) + \mathbb{E}_{s}[\ln p(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{\mu},\boldsymbol{\Lambda})]. \tag{5.24}$$

We now introduce Equation 5.7 utilising the projection form of the model and the variational approximation to the logistic function:

$$\begin{aligned}\ln\left(P(\boldsymbol{y}|\beta,\boldsymbol{z})\right) &\geq \ln h(\beta,\boldsymbol{z},\boldsymbol{\xi}),\\ &= \sum^{i}\sum^{j}\frac{y_{ij}(\beta+\boldsymbol{z}_i\boldsymbol{u}_j^T)}{2} - \lambda(\xi_{ij})(\beta^2+2\beta\boldsymbol{z}_j\boldsymbol{u}_i^T+\boldsymbol{z}_i\boldsymbol{u}_j^T\boldsymbol{u}_j\boldsymbol{z}_i^T)\\ &+ \sum^{i}\sum^{j}\left(\ln\sigma(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2\right),\end{aligned} \tag{5.25}$$

where again ξ is a variational parameter that must be optimised for each link and $\boldsymbol{u}_j$ is the unit vector corresponding to $\boldsymbol{z}_j$. In order to estimate the variational distributions we substitute the distributions in equations 5.16-5.19 into 5.21-5.24 where appropriate and evaluate the expectations. We start with the latent positions $\boldsymbol{Z}$.

$$\begin{aligned}\ln q^*(\boldsymbol{z}_i) &= \mathbb{E}_{\beta,u_j}[\ln p(\boldsymbol{y}|\boldsymbol{z}_i,\beta)] + \mathbb{E}_{K,\mu,\Lambda}\ln\left[p(\boldsymbol{z}_i|K,\boldsymbol{\mu},\boldsymbol{\Lambda})\right],\\ \ln q^*(\boldsymbol{z}_i) &= \mathbb{E}_{\beta,u_j}\left[\sum^{j}\frac{y_{ij}(\beta+\boldsymbol{z}_i^T\boldsymbol{u}_j)}{2} - \lambda(\xi_{ij})(\beta^2+2\beta\boldsymbol{z}_i^T\boldsymbol{u}_j+\boldsymbol{z}_i\boldsymbol{u}_j^T\boldsymbol{u}_j\boldsymbol{z}_i^T)\right]\\ &+ \sum^{j}\left(\ln\sigma(\xi_{ij}) - \frac{\xi_{ij}}{2} + \lambda(\xi_{ij})\xi_{ij}^2\right)\\ &+ \mathbb{E}_{s,\mu,\sigma}\sum^{K}s_{ik}\{\ln\pi_k - \ln\frac{|\boldsymbol{\Lambda}_k^{-1}|}{2} - \frac{1}{2}(\boldsymbol{z}_i-\boldsymbol{\mu}_k)(\boldsymbol{\Lambda}_k)(\boldsymbol{z}_i-\boldsymbol{\mu}_k)^T\},\end{aligned} \tag{5.26}$$

$$\begin{aligned}\ln q^*(\boldsymbol{z}_i) = &\sum_j^n y_{i,j}\left(\frac{\mathbb{E}_\beta[\beta] + \boldsymbol{z}_i\mathbb{E}_u[\boldsymbol{u}_j^T]}{2}\right)\\ &- \sum_j \lambda(\xi_{ij})\left(\mathbb{E}_\beta[\beta^2] + 2\mathbb{E}_\beta[\beta]\boldsymbol{z}_i\mathbb{E}_u[\boldsymbol{u}_j^T] + \boldsymbol{z}_i^T\mathbb{E}_u[\boldsymbol{u}_j^T\boldsymbol{u}_j]\boldsymbol{z}_i\right)\\ &+ \sum^G \mathbb{E}_K[k_{ig}]\{\mathbb{E}_\pi[\ln \pi_g] - \mathbb{E}_\Lambda[\ln \frac{|\Lambda_k^{-1}|}{2}] - \frac{1}{2}\mathbb{E}_{\mu,\Lambda}[(\boldsymbol{z}_i - \boldsymbol{\mu}_k)^T(\Lambda_k)(\boldsymbol{z}_i - \boldsymbol{\mu}_g)]\}.\end{aligned}$$

Now dropping all terms independent of $\boldsymbol{z}_i$ we have:

$$\begin{aligned}\ln q^*(\boldsymbol{z}_i) &= \sum_j^N y_{i,j}\left(\frac{\boldsymbol{z}_i\mathbb{E}_u[\boldsymbol{u}_j^T]}{2}\right) - \sum_j \lambda(\xi_{ij})\left(2\mathbb{E}_\beta[\beta]\boldsymbol{z}_i\mathbb{E}_{u_j}[\boldsymbol{u}_j^T] + \boldsymbol{z}_i\boldsymbol{\Omega}_j\boldsymbol{z}_i^T\right)\\ &+ \sum^K \mathbb{E}_s[s_{ik}]\{\frac{1}{2}\mathbb{E}_{\mu,\Lambda}[(\boldsymbol{z}_i - \boldsymbol{\mu}_k)(\boldsymbol{\Lambda}_k)(\boldsymbol{z}_i - \boldsymbol{\mu}_k)^T]\},\\ &= \boldsymbol{z}_i\left(\sum_j^N \frac{y_{i,j}\mathbb{E}_{u_j}[\boldsymbol{u}_j]}{2} - 2\lambda(\xi_{ij})\mathbb{E}_\beta[\beta]\mathbb{E}_{u_j}[\boldsymbol{u}_j]\right) - \sum_j^n \lambda(\xi_{ij})\boldsymbol{z}_i^T\boldsymbol{\Omega}_j\boldsymbol{z}_i\\ &+ \sum^K \mathbb{E}_s[s_{ik}]\{-\frac{1}{2}\boldsymbol{z}_i^T\mathbb{E}_\Lambda[(\boldsymbol{\Lambda}_k)]\boldsymbol{z}_i + \boldsymbol{z}_i^T\mathbb{E}_\Lambda[(\boldsymbol{\Lambda}_k)]\mathbb{E}_\mu[\boldsymbol{\mu}_k]\},\end{aligned}$$

where

$$\boldsymbol{\Omega}_j = \mathbb{E}_u[\boldsymbol{u}_j^T\boldsymbol{u}_j] = \text{diag}(\boldsymbol{u}_j^T\boldsymbol{u}_j + \sigma_j^2).$$

At this point if we assume a hard clustering framework then the contribution from the mixture model is reduced to simply the component responsible for $\boldsymbol{z}_i$ and we have a distribution for $q^*(\boldsymbol{z}_i)$ that follows a Gaussian distribution:

$$\begin{aligned}\ln q^*(\boldsymbol{z}_i) = &\boldsymbol{z}_i\left(\sum_j^n \frac{y_{i,j}\mathbb{E}_u[\boldsymbol{u}_j]}{2} - 2\lambda(\xi_{ij})\mathbb{E}_\beta[\beta]\mathbb{E}_u[\boldsymbol{u}_j]\right) - \frac{1}{2}\sum_j^N 2\lambda(\xi_{ij})\boldsymbol{z}_i\boldsymbol{\Omega}_j\boldsymbol{z}_i^T\\ &- \frac{1}{2}\boldsymbol{z}_i^T\mathbb{E}_\Lambda[(\boldsymbol{\Lambda})]\boldsymbol{z}_i + \boldsymbol{z}_i^T\mathbb{E}_\lambda[\frac{1}{\Lambda}]\mathbb{E}_{\mu_k}[\boldsymbol{\mu}_k].\end{aligned}$$

with parameters:

$$\boldsymbol{\Sigma}_{z_i}^{-1} = \mathbb{E}_{\Lambda}[\Lambda^{-1}] + \sum_{j}^{n} 2\lambda(\xi_{ij})\boldsymbol{\Omega}_j,$$

$$\boldsymbol{\mu}_{z_i} = \Sigma_{z_i}\left(\sum_{j}^{N} \frac{y_{i,j}\mathbb{E}_u[\boldsymbol{u}_j]}{2} - 2\lambda(\xi_{ij})\mathbb{E}_{\beta}[\beta]\mathbb{E}_u[\boldsymbol{u}_j] + \mathbb{E}_{\mu}[\boldsymbol{\mu}_k]\right).$$

From the structure of the model in Figure 5.5 we observe that the variational update for β will be the same as in the latent position model and follows a Gaussian distribution with mean and variance given by Equations 5.8 and 5.9. It can also be seen from the DAG that the estimation of the remaining parameters in the model is equivalent to inferring a variational Mixtures of Gaussians over the positions $\boldsymbol{z}$. We therefore utilise the standard variational updates provided in [14] to obtain the following equations for the remaining parameters starting with the cluster labels:

$$q^*(\boldsymbol{s}) = \prod_{n=1}^{N}\prod_{k=1}^{K} r_{nk}^{s_{nk}},$$

where

$$\begin{aligned} r_{nk} &= \frac{\rho_{nk}}{\sum_{k=1}^{K}\rho_{nk}}, \\ \ln \rho_{nk} &= \mathbb{E}[\ln \pi_k] + \frac{1}{2}\mathbb{E}[\ln|\Lambda|] - \frac{p}{2}\ln 2\pi \\ &\quad - \frac{1}{2}\mathbb{E}_{\mu_k,\Lambda_k}[(\boldsymbol{z}_n - \boldsymbol{\mu}_k)^T\Lambda_k(\boldsymbol{z}_n - \boldsymbol{\mu}_k)]. \end{aligned}$$

For the discrete distribution $q^*(S)$ we have the standard result

$$\mathbb{E}[s_{nk}] = r_{nk},$$

from which we see that the quantities r_{nk} are playing the role of responsibilities. The optimal solution for $q^*(\boldsymbol{s})$ depends on moments evaluated with respect to the distributions of other variables and so the variational update equations are coupled and must be solved iteratively.

It is convenient for the remaining distributions to define three statistics of the latent positions evaluated with respect to the responsibilities given by [14]:

$$\begin{aligned} N_k &= \sum_{n=1}^{N} r_{nk}, \\ \bar{\boldsymbol{z}}_k &= \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} \boldsymbol{z}_n, \\ s_k &= \frac{1}{N_k} \sum_{n=1}^{N} r_{nk} (\boldsymbol{z}_n - \bar{\boldsymbol{z}}_k)(\boldsymbol{z}_n - \bar{\boldsymbol{z}}_k)^T. \end{aligned}$$

The mixing proportions π_k follow a Dirichlet distribution with parameters given by

$$q^*(\boldsymbol{\pi}) = \text{Dir}(\boldsymbol{\pi}|\alpha).$$

where $\boldsymbol{\alpha}$ has components α_k given by:

$$\alpha_k = \alpha_0 + N_k.$$

The updates for the means and precisions of the components are coupled and we have:

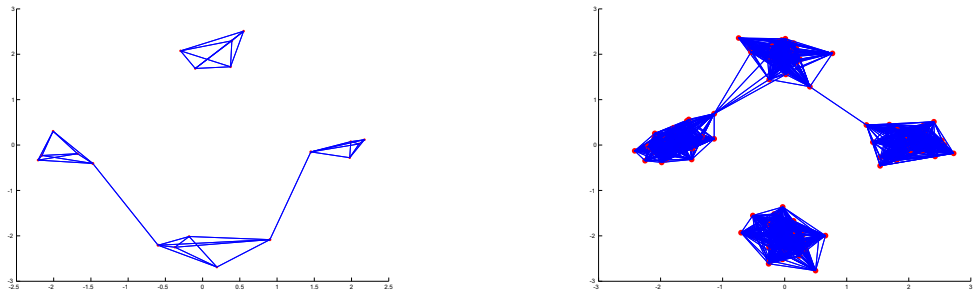
$$q^*(\boldsymbol{\mu}_k, \boldsymbol{\Lambda}) = \mathcal{N}(\boldsymbol{\mu}_k|\boldsymbol{m}_k, (\omega_k \Lambda_k))\mathcal{W}(\Lambda|W_k, \nu_k).$$

where we have defined

$$\begin{aligned} \omega_k &= \omega_0 + N_k, \\ m_k &= \frac{1}{\omega_k}(\omega_0 m_0 + N_k \bar{z}_k), \\ W_k^{-1} &= W_0^{-1} + N_k S_k + \frac{\omega_0 N_k}{\omega_0 + N_k}. \end{aligned}$$

5.5.1 Experiments

The LPCM estimated by variational approximation as described in the previous section is now applied to synthetic and real data. For these experiments we will be required to supply



(a) $n_k = 5$ (b) $n_k = 30$

Figure 5.6: Simulated networks with $K = 4$ and the model correctly allocates the points to their respective clusters in both cases.

the value of K and extension to incorporate inference over K is an avenue for future work to be discussed in Section 7.1. To begin with synthetic data is generated from the LPCM with $K = 4$. For the real data we will use a standard data set from social network analysis - the Sampson's monks dataset which will be described in more detail later.

Data from the Model

Two sets of data were generated from the LPCM model with $K = 4$ and $n_k = \{5, 30\}$. The clusters are of equal size and distributed in a cross shape as shown in Figure 5.6. The nodes are densely connected within a cluster with one disconnected from the rest of the network, highlighting one of the advantages of our approach over those that are restricted to fully connected graphs. The model estimated by the variational method from the corresponding adjacency matrix correctly clustered the nodes into their distinct groups. A small value of $n_k = 5$ was chosen to generate a data set comparable in size with the Sampson data set studied in the next section while the larger data set with clusters of size $n_k = 30$ produces a network that would require a long runtime using MCMC methods but is easily manageable using the variational approach.

Sampson's Monk Data

The well studied Sampson's monk dataset [115] was used in both the original LPCM paper [58] and the initial variational solution [114] and thus we choose to use it here to allow comparison of results. The data stem from an ethnographic study of community structure in a New England monastery. The study describes several social relations among a group of men (novices) who were preparing to join a monastic order. We only consider the 'liking' social relationship, at five different time points the novices were asked to indicate whom they liked the most and to provide a first, second and third choice. The study captures a period of great turmoil and division and shortly after the fourth measurement four of the novices were expelled. To generate the network we follow the method used in [58], only answers from the first three interviews are considered and any indication of 'liking' is taken as an edge between two individuals, the 'dislike' relation is not considered. Sampson provided a description of the clustering based on information that was collected at the end of the study period. He identified three main groups: the Young Turks (seven members), the Loyal Opposition (five members) and the Outcasts (three members). The other three monks wavered between the Loyal opposition and the Young Turks which he described as being in intense conflict [115]. With this in mind we initially began our experiments with the value of $K = 4$ however the results obtained were highly unstable and generally resulted in two empty clusters. The value of $K = 2$ was then chosen and a stable solution obtained which is shown in Figure 5.7, where the allegiance to the different factions is indicated using a letter for each node. The results show clear separation of the data into two clusters. The clusters are defined by the two opposing factions, the Young Turks and the Loyal Opposition, with two members of the waverers joining the opposition and one the Young Turks. The Outcasts are absorbed into the Young Turks which is the one negative as the MCMC method in [58] was able to separate these novices into their own cluster to produce a $K = 3$ solution. The central position of the third 'waverer' is interesting and justifies the name ascribed to this particular group of novices as the node has links to both factions which defines its position. While the embedding in Figure 5.7 shows the Outcasts somewhat separated from the Turks they do not form their own cluster and it could be that, as we have remarked previously, we are suffering from a lack of data, which has a negative

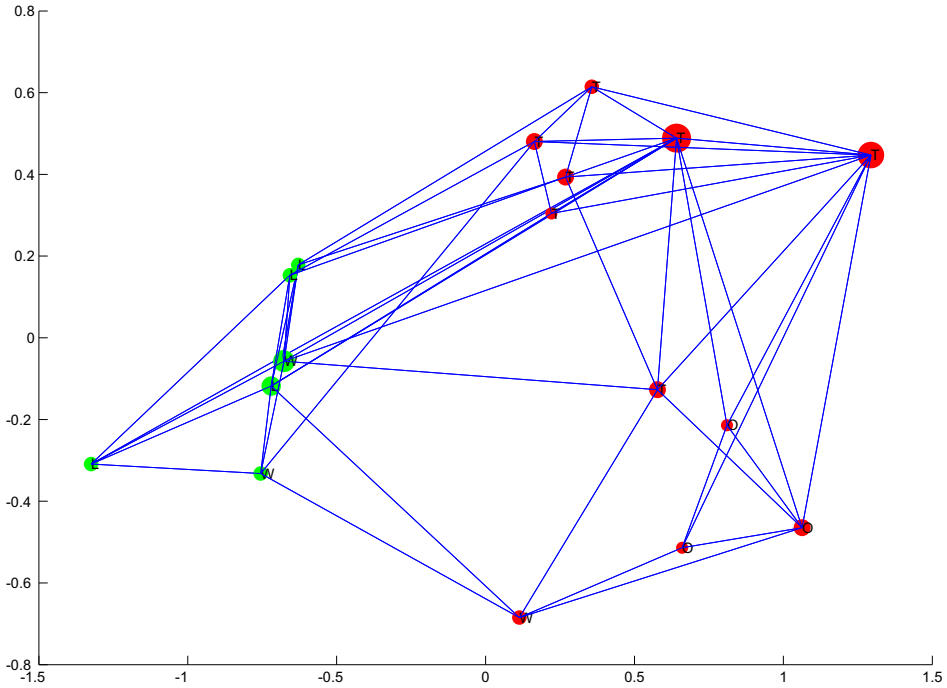


Figure 5.7: Results obtained using the variational approximation method to estimate the LPCM with $K = 2$ for Sampson's monk data. The colours of the nodes signify cluster membership and the letters above each node indicate which of the four factions the corresponding person was affiliated with, according to [115].

impact on the performance of variational methods. In the main however the results obtained agree with those found in the original paper estimating the LPCM using MCMC [58].

5.6 Conclusion

Chapters 4 and 5 both are devoted to improving the model based embedding approach for network data. This chapter concentrated on the development of a method for estimating the Latent Position and LPCM models by variational approximations. This required the use of an alternate phrasing of the model using the cosine angle distance and a local variational approximation to the logistic function. After calculating analytical update equations for the model parameters the models were deployed on a number of synthetic networks and the

LPCM was tested on a benchmark dataset. The synthetic networks used were significantly larger than those tackled with the MCMC approach. The performance on the Sampson's monks dataset demonstrated the method's ability to correctly capture the major cluster structure in the data and closely matches that obtained by the original MCMC method in [58] with the exception of the 'Outcasts' cluster which were not separated into their own group but assigned to the larger 'Young Turks' cluster.

The method developed is a more satisfactory solution to the problem of estimating the LPCM by variational approximations than that of previous efforts. However this comes at the cost of a fixed embedding dimension of $p = 2$, due to the use of the von Mises distribution. As mentioned the use of the Fisher-Bingham distribution is one potential solution that could be explored in future work. Inference over the value of K is another aspect that could be extended following the same method as those seen in Chapter 3 with the MFA model estimated by variational approximations [10]. These are two aspects in which the variational approach could be extended and that for the time being lead us to favour the BDMCMC approach developed in Chapter 4. In the next chapter we will use this framework to develop a method for detecting changes in graphs that unlike current state of the art methods does not monitor specific features of the graph. Instead we utilise Canonical Correlation Analysis to compare successive embeddings of networks.

Chapter 6

A Latent Variable Framework for Comparing Networks

The Latent Position Cluster Model, studied in Chapters 4 and 5, provides us with a fully Bayesian model for embedding graph data. Extension of the estimation method to incorporate inference on K, the number of mixture components in the model, allows for fully unsupervised community detection of a network. An advantage of a model-based approach is that it allows for further analysis and interrogation of the model. For example the inference on K can in itself be used as a means of detecting changes in a network, albeit only those that manifest as simple changes in the community structure. As we demonstrated using the VAST challenge data in Section 4.5.2 this can be insufficient to capture more subtle differences between networks and in this chapter we propose a novel method utilising the LPCM representation that allows comparison of networks to detect further differences in network structure. The main strength of the method is its generality as there is no a priori selection of any particular graph metric or feature as a means of defining change. This distinguishes it from other state of the art methods for mining graph data. Another key distinction is that due to the nature of the LPCM, we are essentially analysing a vector representation of the network which allows us to deploy standard multivariate analysis techniques rather than structural pattern recognition methods associated with graph data.

The chapter proceeds as follows, in Section 6.1 we review the current literature and comparable methods for detecting changes in graphs. This is a rapidly developing field with

some of the earliest papers from just a decade ago containing only a handful of references. Section 6.2 then describes the proposed method for comparing networks. This is based on Canonical Correlation Analysis which was introduced earlier in Section 2.1.3. Change detection requires the use of a test statistic to allow confidence intervals to be set and we derive one based on the transformation matrices associated with CCA in Section 6.3. In Section 6.4 a number of experiments are conducted using synthetic data to assess the performance of the method under known conditions and then finally applied to real data in the form of mobile phone data from the VAST challenge and email communication data from the Enron Email Corpus.

This chapter features expanded versions of material published in [99] and reproduced with permission from the IEEE, ©2012.

6.1 Review of Change Detection in Networks

The field of statistical change detection is an area that has been extensively researched over the years with a vast catalogue of literature behind it. By comparison change detection in a graph setting is an under researched area although the number of papers in the field is increasing rapidly from just a handful a decade ago. The applications of graph based change detection are as numerous as the use of graphs themselves. Network intrusion detection for computer networks [96], credit card fraud detection [4], drug trafficking [34], epilepsy seizure onset [79] and social network analysis [87] have all made use of such methods. Graph based change detection methods can broadly be divided into two categories, methods for detecting changes or anomalies *within* a graph and methods for detecting changes *between* a sequence of graphs. The method we have developed is for detecting structural differences between graphs however there is a great deal of crossover between both objectives with methods and principles from each being applied in both domains. With this in mind we will give an overview of both approaches starting with anomaly detection within a graph.

For a graph based anomaly there are several situations that may occur, including:

1. A node exists that is unexpected.

2. An edge exists that is unexpected.
3. The node label is different than expected.
4. The edge label is different than expected.
5. An expected node is absent.
6. An expected edge between vertices is absent.

There are three general categories of anomalies: insertions(1,2), modifications(3,4) and deletions(5,6) [34]. Anomalies can be individual nodes themselves or anomalous substructures within the graph. The dominant theme or method in this area is the use of Minimum Description Length (MDL) based algorithms. The MDL is the lowest number of bits needed to encode a piece of data [96]. Methods based on this heuristic iteratively cycle through the graph finding the most common substructure and compressing these substructures into new nodes, anomalies can then be detected as substructures with a high compression cost that occur infrequently. This approach is similar to the graph matching approach utilised in structural anomaly detection or detecting changes between graphs and has been used in a number of papers for network intrusion detection [96, 18, 34] and drug trafficking detection [34].

A property that is inherent in graphs is the presence of graph spectra. Analysis of the eigenvalues of the graph Laplacian of the adjacency matrix can be used for graph clustering [136]. This spectral analysis approach can also be applied to anomaly detection and analysis of the principal eigenvectors of the graph was used to detect anomalous nodes in [68] as part of a network monitoring system.

The extraction of features such as the spectra above can be extended to incorporate some of the infinitely many other possible graphs features. In [4] specific features of a node's local neighbourhood, referred to as an *egonet*, are extracted including the principal eigenvalues. The relationship between pairs of the specifically selected features are shown to follow a power law and anomalous nodes can be picked out as those deviating from this relationship.

The egonet method has also been applied to detect changes in a time series of mobile communication graphs [3]. This involved the extraction of a set of 12 features for each node

in the network. The correlations between these attributes for each node are monitored and deviations from this correlation structure indicates a change in the network. The method effectively works in a multivariate space rather than the graph space and it appears to be quite similar to the approach we have taken except rather than extracting particular features we embed the network in a low dimensional multivariate space using the LPCM. Indeed many of the features selected such as the degree, number of neighbours, number of triangles etc. influence the positions of the nodes in the LPCM.

A network feature extraction approach has also been employed to detect changes in social networks [87, 86]. The authors monitored two specific node features, betweenness centrality, a measure of how often a node lies along the shortest path between two other nodes for all other nodes in a graph, and closeness centrality which measures the sum of the distances from a particular node to all other nodes in the graph. An overall network feature, the density was also monitored. The density of a network is the measure of how many links exist in the graph divided by the total possible number of links. Organisations with high density are well connected internally. The key issue with this approach and the egonet method described above is that there are hundreds of different network measures that can be calculated from the entire graph or for individual nodes. Selecting a specific subset of features to monitor for changes a priori implicitly limits the types of changes that can be detected as to just those that produce changes in those features. For example it is possible for the structure of a network to change while the overall density of the network remains the same simply by maintaining the same number of nodes and edges.

An alternative approach that has been used widely for comparing graphs for either classification or change detection is the graph matching approach [18, 27]. The most stringent form of graph matching is graph isomorphism, where a one-to-one correspondence must be found between each node of the graphs. A weaker form of matching is subgraph isomorphism, that requires that an isomorphism holds between one of the two graphs and a node induced subgraph of the other. Finally the maximum common subgraph approach maps a subgraph of the first graph to an isomorphic subgraph of the second one, since such a mapping is not uniquely defined, the goal is to find the largest subgraph for which such a mapping exists. Graph edit distance [116] can be used as a measure of the distance or dissimilarity between two graphs. In graph edit distance computation, one applies a sequence

of edit operations on the two given graphs so as to make the first graph identical, or isomorphic, to the second one. The length of the shortest edit sequence of this kind is defined as the edit distance of the two graphs under consideration. Often a cost is assigned to each edit operation. In this case, edit distance is defined as the cost of the cheapest sequence of edit operations that make the two graphs isomorphic [18]. This graph edit distance can be used as a measure of change in a sequence of graphs however there is a high computational complexity associated with computation

6.2 Comparison of Networks using Canonical Correlation Analysis

Applying the LPCM to network data produces a q-dimensional vector representation of the network embedded in Euclidean space. In Section 2.1.3 Canonical Correlation Analysis (CCA) [7, 65, 137] was introduced as a latent variable model and a Bayesian treatment described. The DAG for this model is shown in Figure 6.1 illustrating the dependencies in the model. CCA is a latent variable model that gives a measure of the correlation between two sets of multivariate data.

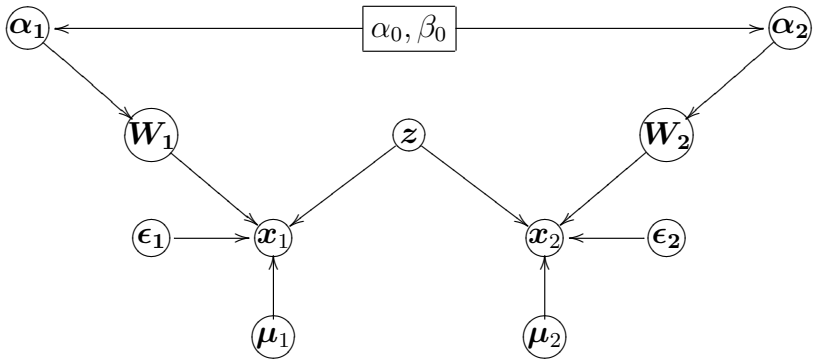


Figure 6.1: Directed Acyclic Graph of the Bayesian CCA model. The correlations between data sets $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ are explained by the shared latent variable $\boldsymbol{Z}$.

The embedding representation obtained from the LPCM can be used in conjunction withs CCA to compare two networks. The canonical weights, $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$, obtained from CCA, describe the linear transformations required to map the network embeddings $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ to a common latent variable $\boldsymbol{Z}$. These weight matrices provide a means of comparing the networks in that it is assumed similar networks will share similar mappings. To test this

assumption we construct an identity:

$$\boldsymbol{W}_1^{-1} \times \boldsymbol{W}_2 = \boldsymbol{I}_q, \tag{6.1}$$

which allows us to compare two networks, this is one possible choice but we note that other variations are possible too. Networks that obey this identity are perfectly matched whereas for networks which violate this identity there will be a difference in structure. It is convenient at this point to appeal to a frequentist approach and we construct a hypothesis test with the null hypothesis that the two networks are the same. However the test statistic is matrix valued and the next section describes how to construct confidence intervals for the elements of the matrix which are then tested individually.

6.3 Test Statistic for Canonical Weights

Using the latent variable model formulation of CCA from Section 2.1.3, if the data $\boldsymbol{X_1}$ and $\boldsymbol{X_2}$ are highly correlated then the translation to the latent variable space should respect the identity in Equation 6.1. In order to determine whether the networks are different we need to determine the distribution of this matrix-valued test statistic and whether a value falls outside the expected range given the distribution. This could be tackled using methods from random matrix theory [35] however for simplicity we choose to make inference about each element of the matrix separately. Also as a consequence of the Bayesian approach to model estimation outlined in Section 2.1.3 which we will follow, the ARD priors discussed will drive the columns of $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ to zero if there are no correlations in the data along this dimension. This gives us a second flag to detect changes however we note that the test statistic will capture this indicator as a large spike in the test values caused by inverting a value close to zero.

The prior distribution of each column of the canonical weights matrix, $\boldsymbol{W}$, is an independent Gaussian distribution and utilising the conjugate prior distributions set out in Section 2.1.3 in Equations 2.11 and 2.14 it is possible to calculate the posterior distribution of each column to also be a Gaussian. Full posterior distributions for the model can be found in [76, 135]

If we partition the joint Gaussian appropriately we can obtain the marginal distribution for each element of the column:

$$w_{ij} \sim \mathcal{N}(\hat{w}_{ij}, \hat{\sigma}_i),$$

where $\hat{w_{ij}}$ is the posterior expectation and $\hat{\sigma}_i$ is the posterior variance. We restrict ourselves to the simplest case, $q = 2$, for which we obtain the test matrix:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} \times \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \frac{1}{|\boldsymbol{W}|} \begin{pmatrix} de - bg & df - bh \\ -ce + ag & -cf + ah \end{pmatrix},$$

where each element is given by a sum of products of Gaussians. The next section describes a method to calculate the distribution of such a sum.

6.3.1 Products of Gaussians

The distribution of a sum of products of Gaussians is not available in closed form. Thus we will approximate the distribution via a set of moments. The moment generating function, $M_z(t)$, for a product of Gaussians was found by [29] who calculated it to be:

$$\begin{aligned} M_z(t) =& (\sqrt{[1-(1+\rho)t][1+(1-\rho)t]})^{-1} \\ &\times e^{\frac{2\delta_1\delta_2 t + (\delta_1^2+\delta_2^2-2\rho\delta_1\delta_2)t^2}{2[1-(1+\rho)t][1+(1-\rho)t]}} \end{aligned} \tag{6.2}$$

where $z = \frac{x_1 x_2}{\sigma_1 \sigma_2}$, $x_1 \sim \mathcal{N}(\mu_1, \sigma_1)$, $x_2 \sim \mathcal{N}(\mu_2, \sigma_2)$ and $\delta_1 = \mu_1/\sigma_1$. By manipulating $M_z(t)$ we can find as many moments as required. These moments can be used to calculate the mean, and variances of the product of two Gaussian variables which are:

$$E(z) = \mu_1\mu_2 + \rho\sigma_1\sigma_2,$$

$$V(z) = \mu_1^2\sigma_2^2 + \mu_2^2\sigma_1^2 + \sigma_1^2\sigma_2^2 + 2\rho\mu_1\mu_2\sigma_1\sigma_2 + \rho\sigma_1^2\sigma_2^2,$$

A further simplifying assumption is needed. In the case when $\rho = 0$, x_1 and x_2 are

independent, the mean and variance reduce to:

$$E(z) = \mu_1\mu_2,$$
$$V(z) = \mu_1^2\sigma_2^2 + \mu_2^2\sigma_1^2 + \sigma_1^2\sigma_2^2,$$

which are simple to evaluate once the moments of the original variables are known. The assumption of independence between the variables presents another advantage. Although Equation 6.2 shows that the analytic result of the product of two Gaussian distributions is not a Gaussian distribution, in the case of independence the limit, as δ_1 and δ_2 approach infinity, of $M_z(t)$ is a Gaussian distribution. In other words, the product of $x_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$ and $x_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$ tends towards a $\mathcal{N}(\mu_1\mu_2, \mu_1^2\sigma_2^2 + \mu_2^2\sigma_1^2 + \sigma_1^2\sigma_2^2)$ distribution as δ_1 and δ_2 increase [138].

One way to predict the adequacy of the Gaussian approximation is by considering the skewness of z from the analytic result. As the skewness decreases the approximation improves. Using the approximation the sum of products becomes a simple sum of Gaussian distributions which are additive. It should be noted that summing a sufficient number of independent random variables together will produce a Gaussian distribution as per the central limit theorem. Increasing the embedding dimension would result in the addition of more products of Gaussians which may improve the validity of the approximation used [99]

6.3.2 Empirical Analysis of Test Statistic

To verify that the approximation in Section 6.3.1 is satisfactory we generate a Monte Carlo sample of the sums of product of Gaussians by taking 10000 samples from two independent Gaussian distributions and then multiplying and summing individual samples together to get 10000 samples from the corresponding sum of product of Gaussians. A histogram of these samples is shown in Figure 6.2, with the Gaussian distribution with parameters calculated as discussed in Section 6.3.1 overlaid in red. The Kolmogorov-Smirnov(K-S) test [85] was used to evaluate the goodness of fit. This test was conducted 100 times, with

four randomly chosen examples shown in Figure 6.2, using randomly chosen parameters and in all tests we fail to reject the null hypothesis at the 95% confidence level. A more thorough analysis of the approximation can be found in [138].

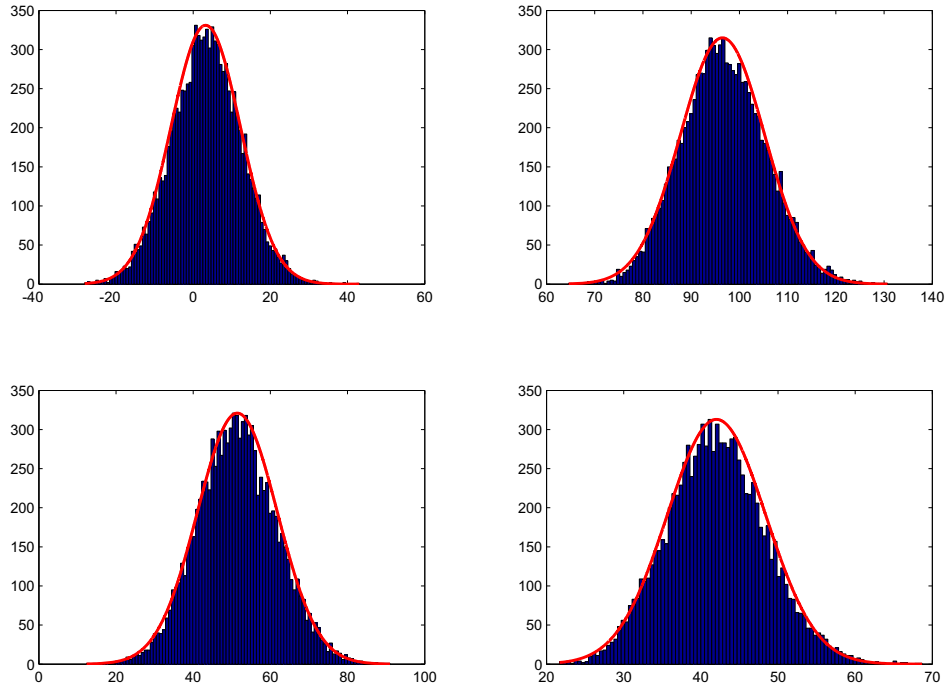


Figure 6.2: Monte Carlo samples of a sum of products of Gaussians in blue with the approximated Gaussian distribution overlaid in red. The samples and approximated distribution were compared using the Kolmogorov-Smirnov test.

6.4 Experiments

This section describes a number of experiments carried out using simulated and real network data to evaluate the ability of our approach to detect a change in network structure. As mentioned in Section 6.3 the embedding dimension is fixed to the simple case of $q = 2$. This value of q is preferred partially for computational simplicity but also because the development of the LPCM favours $q = 2$ for the purpose of visualisation. The identity, Equation 6.1, is evaluated and each element tested at a 95% confidence level, $+/- 1.96\sigma$.

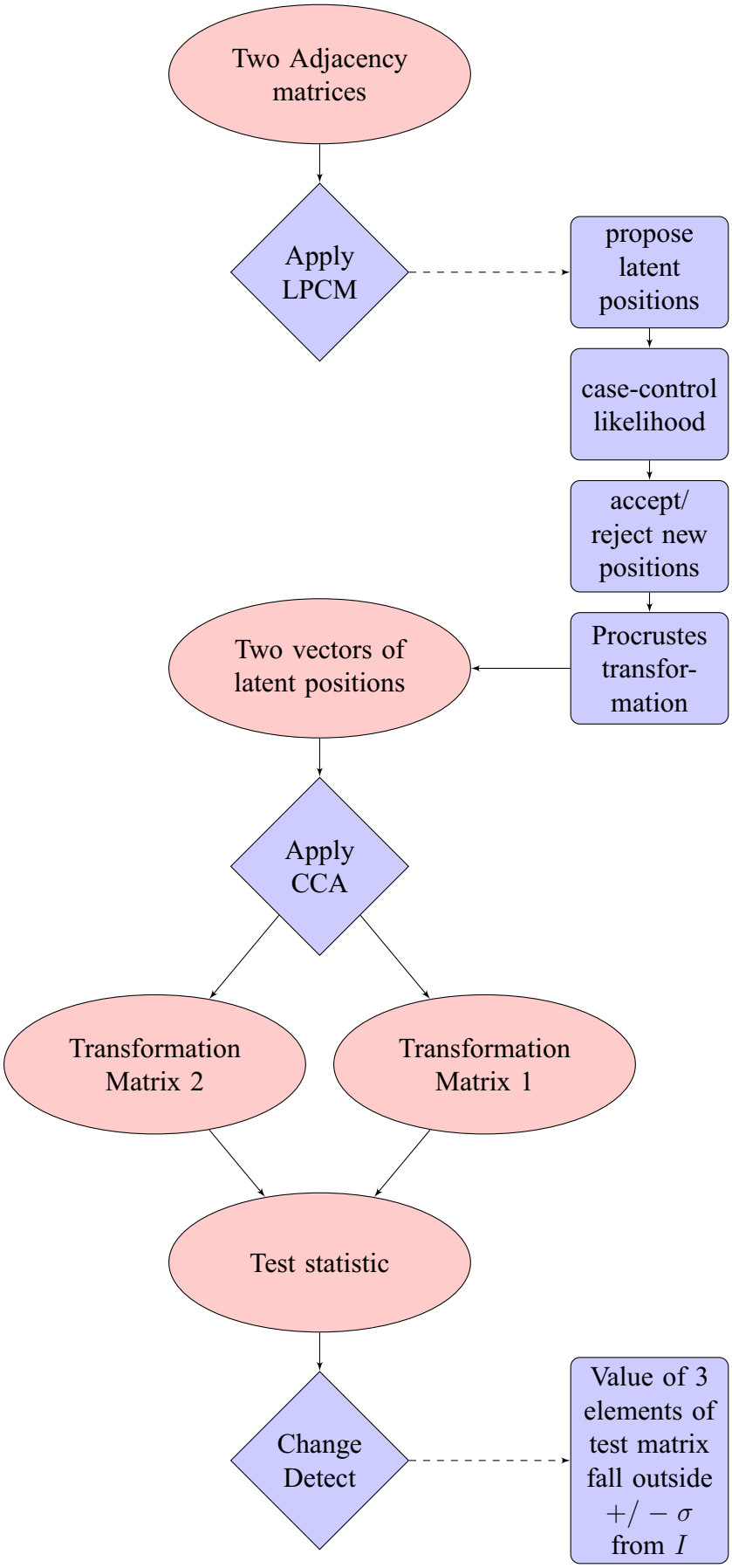


Figure 6.3: Outline of the steps involved in the network comparison method.

With $q = 2$ there are four variables to consider and we deem the networks to be different if the results of at least 3 tests are in agreement. This is perhaps an overly conservative method of combining the information and a weighted voting scheme could be considered in future work as well as a multivariate test comparing posterior distributions over $\boldsymbol{W}$. A schematic outlining the full process for comparing two graphs is outlined in Figure 6.3.

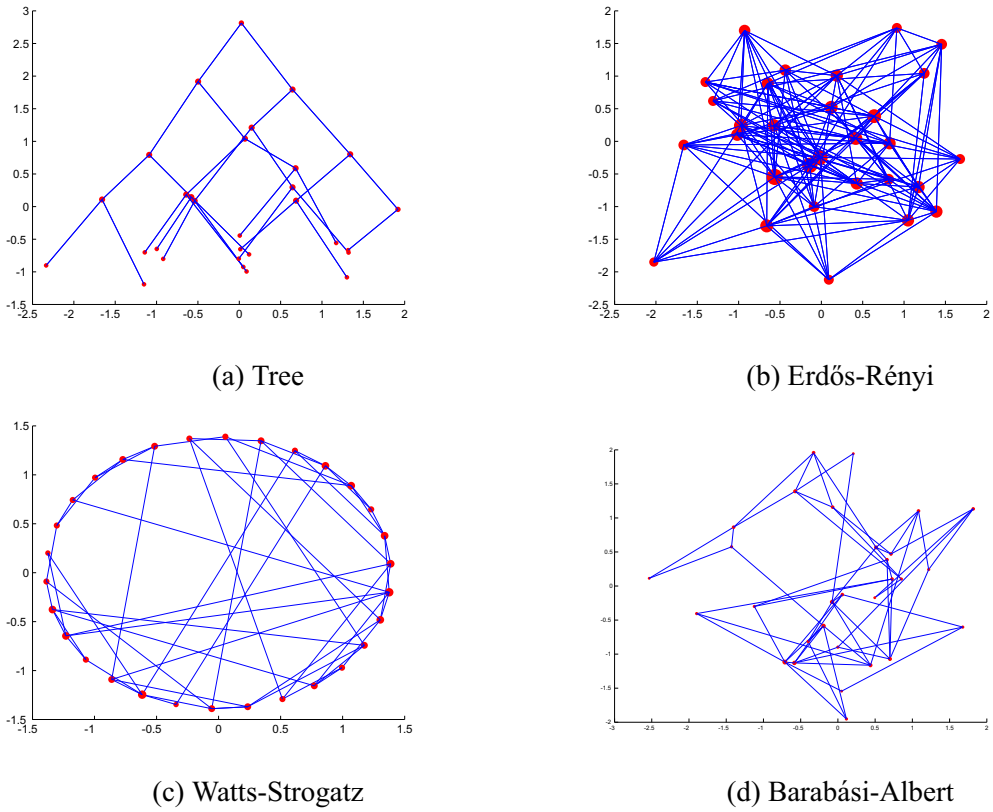
The experiments are organised as follows, using synthetic data generated from the models introduced in Section 4.1 we initially verify that no change is detected when the network is presented with two networks that are the same, we then conduct a change detection experiment comparing networks with completely different architectures. To conclude the synthetic experiments we attempt to gain further insight into the capabilities of the method by quantifying the magnitude of the change that can be detected. This is performed in two ways, first by introducing increasing levels of randomness into the network by randomly rewiring increasing numbers of randomly chosen nodes. As an attempt to introduce more subtle changes into the network a second experiment is performed where the nodes are rewired but the degree of the node remains unchanged. Thus the degree sequence of the graph, which is a generally a sufficient statistic [44] to identify a graph, is unchanged.

To conclude we deploy our method on two datasets, the VAST challenge data encountered previously in Section 4.5.2 and the Enron Email Corpus.

6.4.1 Simulated Networks

In this section the performance of the method is assessed using simulated data. First we verify that the method is consistent in detecting no change and that the identity, Equation 6.1, holds. Networks with different topologies are then compared to demonstrate the ability of the method to detect differences between networks. The data used will be made up of four, non-trivial network topologies. We use a tree network which is a regular network that obeys the LPCM. The other three networks are different network models with specific characteristics described in Section 4.1. A fully random network or Erdős-Rényi graph in which the probability of a link between two nodes is the same for all nodes. A Watts-Strogatz network which is a type of small world network and a Barabási-Albert network which is similar to real world networks demonstrating preferential attachment. Examples

of these networks are shown in Figure 6.4.



(a) Tree

(b) Erdős-Rényi

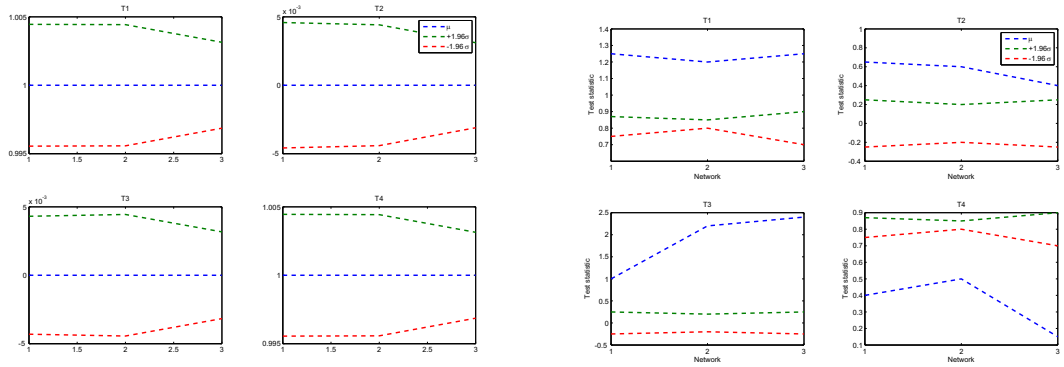
(c) Watts-Strogatz

(d) Barabási-Albert

Figure 6.4: Tree, Erdős-Rényi, Watts-Strogatz and Barabási-Albert networks, $n = 30$, visualised in two dimensions. The networks provide a good evaluation for our method each having different network characteristics, such as average degree and betweeness centrality.

Consistency Experiment

To verify that the method does not detect spurious changes we perform a comparison of successive embeddings of the same network, for each of the four networks in Figure 6.4. As discussed earlier the networks are embedded in two dimensions and the test matrix obtained from Equation 6.1 consists of four elements. The distributions of these individual elements were obtained utilising the approximation to a product of Gaussians outlined in Section 6.3.1 and the means are plotted in Figure 6.5a. Having the distribution of the test statistic elements allows us to calculate confidence intervals corresponding to $+/-1.96\sigma$ these are plotted around the elements of the Identity matrix. A change is said to have



(a) Consistency experiments (b) Different architectures

Figure 6.5: Test statistics obtained for two different experiments. (a): consistency experiment, the combined embedding-CCA method is presented with two adjacency matrices that are the same, no change is detected. (b): comparison of networks with different architectures. A change is detected at each comparison step. The expectation of each of the four elements of the test statistic are plotted in blue and confidence intervals at the 95% margin are plotted in red or green.

occurred if the mean of three elements of the test statistic fall outside these values and Figure 6.5a shows that no change is detected for any of the networks.

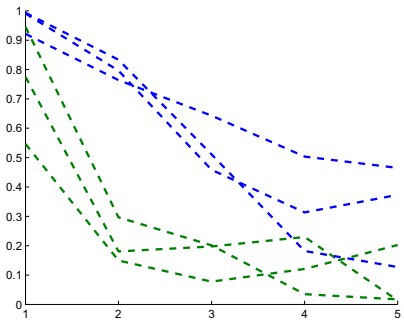
Network Comparison

The previous experiment confirms that Equation 6.1 holds and that no change is detected when the method is presented with two adjacency matrices that are the same. The next step is to test the method's capability to detect differences between adjacency matrices with different structure. The four networks shown in Figure 6.4 are compared and the test statistics obtained are shown in Figure 6.5b. The networks are compared successively starting with the tree and through to the Barabási-Albert network. A value outside the confidence intervals is obtained in at least three of the test statistics in each comparison indicating a change has been detected.

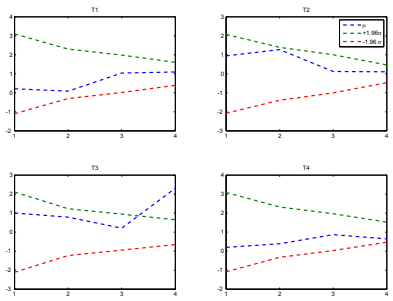
Change Magnitude Experiment

The previous experiments demonstrated the ability of the combined embedding-CCA method to detect a change when there is a large shift in the structure of a network, i.e. from regular to random. To gain further insight into the capabilities of this approach a further set of experiments was performed studying the behaviour of the method for incrementally larger changes to a network. This section presents the results of these experiments where a network, generated from the Latent Position Model, is subjected to increasing amounts of structural change. This is done in two ways: in the first set of experiments nodes are selected at random and then randomly rewired with probability p as in an Erdős-Rényi graph. The number of nodes is increased in stages increasing the magnitude of the change in the network. The second set of experiments follows the same idea however the nodes are rewired so that the degree sequence is preserved. This introduces a change into the network while still preserving some of the original structure, indeed the degree sequence is considered a sufficient statistic for defining a graph [44]. This makes the test a more challenging task for a change detection algorithm as in some sense we are comparing a network that can be considered the same. The original network is generated from the Latent Position model, drawing a set of points from a Gaussian distribution and generating links between the corresponding nodes. A link is drawn as a Bernoulli random variable with probability defined by Equation 4.4 to create the adjacency matrix. The size of the network is varied with $n \in \{10, 25, 50, 100\}$ and the number of nodes randomly rewired is increased in steps of 20% starting from a single node. The canonical correlations, defined in Section 2.1.3, obtained during the random rewiring experiment are shown in Figure 6.6a and show that as the amount of randomness introduced to the network increases the correlation of the embedding with that of the original decreases. The test statistic obtained comparing the original network with the degree rewired networks with $n = 100$ is shown in Figure 6.6b. The test statistics obtained for both rewiring experiments for all n are provided in Appendix B and we now summarise the main features observed. The results obtained from this experiment reveal some interesting characteristics of the method. For both forms of rewiring the points at which a change is detected are the same at:

- $n = 10$ a change is detected at step 3, when 50% of nodes have been changed.



(a) Canonical correlations



(b) Test Statistic $n = 100$ degree rewiring

Figure 6.6: Canonical correlations and test statistics obtained for the change magnitude experiment. (a): canonical correlations for the random rewiring experiment. Each iteration of the network is compared to the original. As the magnitude of the change induced increases the correlation between embeddings is reduced, the first canonical correlation is plotted in blue and the second in green. (b) first four steps of the degree rewiring experiment for $n = 100$, no change is detected with each test statistic value within the 95%confidence intervals.

- $n = 25$ a change is detected at step 3 when 44% of nodes have been changed.
- $n = 50$ a change is detected at step 4 when 62% of the nodes have been changed.
- $n = 100$ a change is detected at step 5 when 81% of nodes have been changed.

Interestingly the results show that the type of change that occurs makes no difference in terms of detection but that the **magnitude of change** is important and as the size of the network increases so does the fraction of the network that needs to change for a detection to occur. Informed by these results the method is applied to data from real networks.

6.4.2 Real Network Data

The data tested so far has consisted of networks that obey a single simple structure. Real networks are much less ordered and to test our method under more realistic conditions we now apply it to two datasets of communication networks. The first is simulated phone call data from the IEEE VAST 2008 Challenge [56] and the second consists of the patterns of emails between employees in the Enron corporation.

VAST Challenge Data

While the data is synthetic it is simulated with sufficient realism that it captures daily cycles in call patterns and can be treated as real. The data consists of ten days of cell phone calls made by 400 unique cell phones. Previous work on the challenge detected anomalous activity beginning on the eighth day of recording [60]. The data is binned into daily calls to generate separate networks for each day. The latent representations of two of these days are shown in Figure 6.8. In this case visual inspection of the embedding is insufficient to suggest a change in network structure. We test over a nine day period leading up to and including the change day. The results are shown in Figure 6.7 where the days are compared consecutively, day 1 with day 2 and day 2 with day 3 etc. A large spike is observed in all four elements of the test matrix on the anomalous day. There is some variability observed throughout the run but the largest change by far is on the anomalous day. According to the results of the VAST challenge [56] at this point three of the major players in the network, i.e. nodes with the largest degrees, switch phones. This produces a massive shift in the network with the original nodes going silent and the new nodes taking their place. This change in the structure is then detected by the embedding-CCA method.

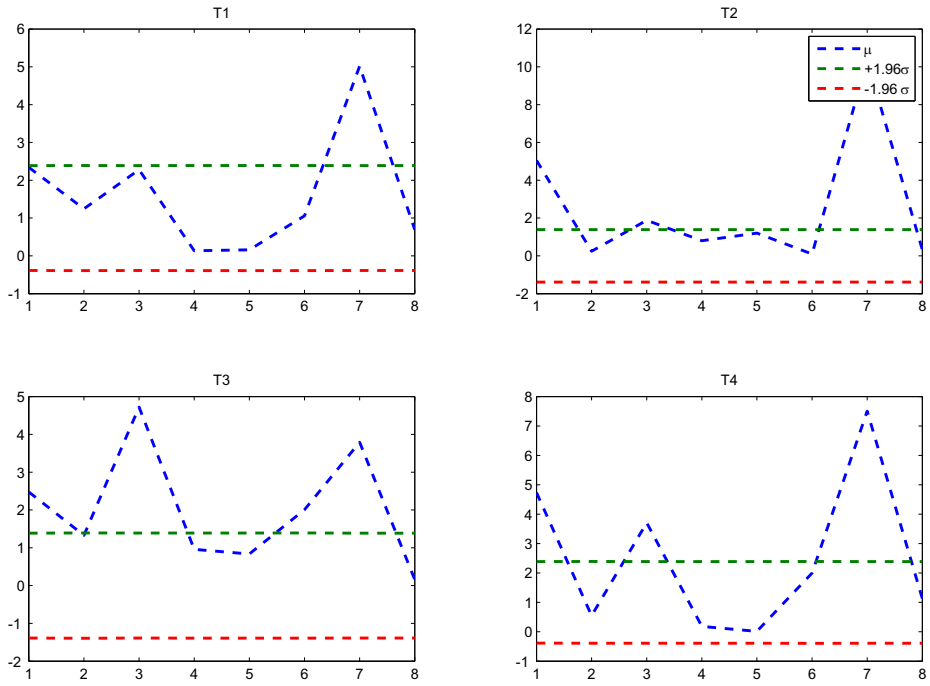
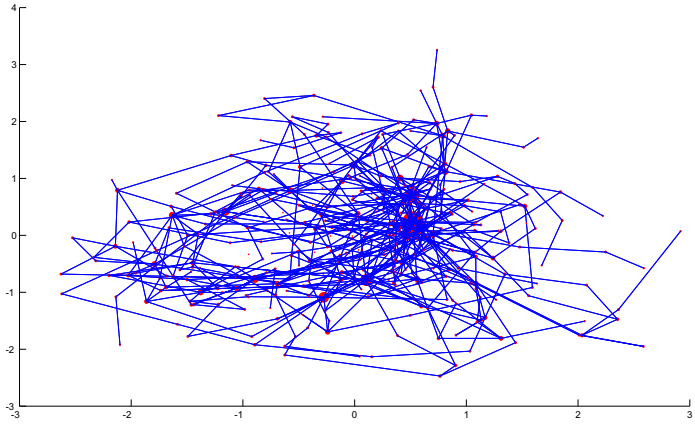


Figure 6.7: Test statistic for the VAST networks comparing each day with the previous. We see an abrupt change on the day when the pattern of activity shifts. The activity in the proceeding days is also shown to be broadly similar with some variability from day to day but not of sufficient magnitude to flag a change. [99] ©IEEE 2012.

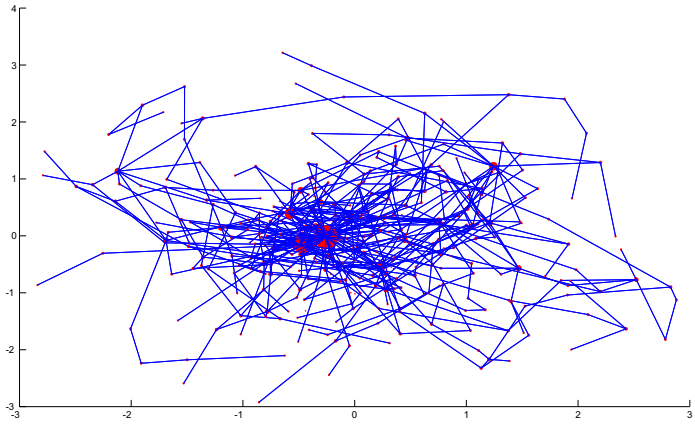
Enron Dataset

The Enron email corpus is one of the largest and best known datasets of real email data. The data was made public by the Federal Energy Regulatory Commision following their investigation into the collapse of the Enron corporation, which at one point had been the seventh largest, by revenue, business organisation in the United States. The data was collected from the email documents of 150 senior executives in the Enron company. The full corpus represents a large collection (about half a million emails) and temporal record of email conversations over a period of 2.5 years [32]. The data has been the subject of a great deal of research for both social network analysis of the communication patterns and natural language processing of the email content.

A consequence of the data being analysed with different scientific objectives is that



(a) Selected day of normal activity



(b) Day of anomalous activity

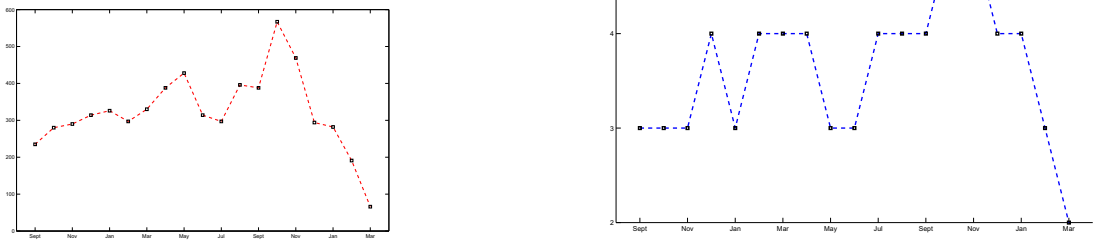
Figure 6.8: Embeddings of VAST network activity on two different days. The network on the right is taken on the anomalous day. The nodes are plotted in red with the size of a node being proportional to the degree of the node. Visual inspection of the graphs is insufficient to determine whether a change has occurred. [99] ©IEEE 2012.

different variants of the original complete data are utilised. For example every email address found in the sent or received folder of one of the 150 employees can be considered as a unique node producing a network with $n = 34,275$ [127]. Alternatively a network of just 150 nodes can be obtained by considering just the communication between the 150 employees who's emails were published [117, 24]. The data can also be analysed using different time slices, on a daily, weekly or monthly basis.

Our analysis concentrates on the original 150 senior executives and only considers the email communications between these individuals. Individuals who send and receive less than 5 emails are removed, reducing the total number of users to 148. The data is aggregated into traffic over a month. The initial months yield low email traffic thus we only consider the time period from September 2000 to March 2002 following the same reasoning of [128] but including more of the earlier months as they [128] only consider the final 11 months. We form an undirected network where the identity of the sender or receiver is irrelevant and an email indicates a link between the two nodes. Figure 6.9a plots the number of unique connections per month in the data and serves to illustrate the overall activity in the network. During this period of time Enron was going through a crisis with a number of significant events occurring such as the resignation of CEO Jeffrey Skilling in August 2001 and the company filing for bankruptcy in December 2001. The hypothesis that motivates our analysis is that during a crisis the normal communication patterns within a large organisation may change and we seek to detect any significant differences between consecutive months.

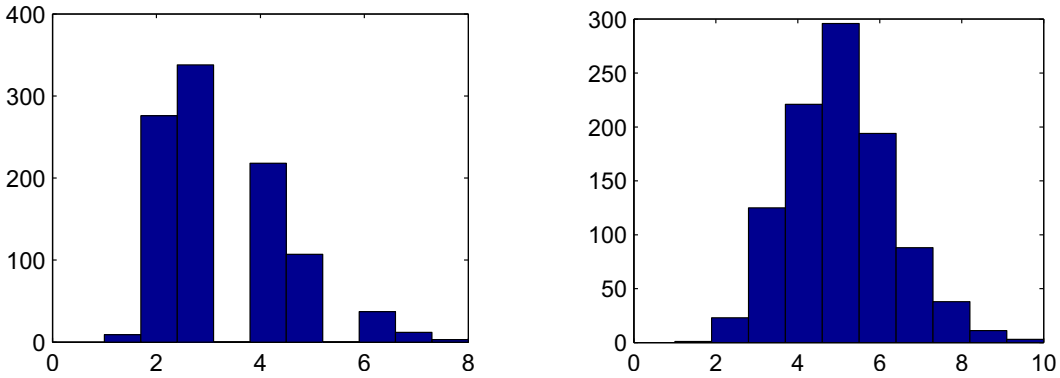
The first step in analysis was to examine the community structure of the network to determine whether employees were separated into distinct clusters or if there was a more integrated community structure. The LPCM was deployed using BDMCMC to estimate K, as described in Section 4.5. The label switching problem associated with applying MCMC to mixture models is resolved using the method outlined in Section 3.2.4. The sampler was run for a burnin period of $10,000$ samples and then a further $10,000$ samples were drawn with every 10^{th} retained. The results obtained for each month are shown in Figure 6.9b where the medians of the posterior distributions of K are plotted. The posterior distribution of K for September 2000 and October 2001 are shown in Figure 6.10. The distributions are both peaked at a single value however there is still significant support for a number of other configurations. Table 6.1 shows the numbers of employees in each cluster for each month, these are calculated as the mean of the posterior distribution of the mixing proportions, π_k for $k = 1, \ldots, K$.

From Figure 6.9b we can see immediately that the community structure of the network changes over time with $K = 4$ being the most frequent arrangement. These changes in K could themselves be used as indicators of change in the network however the differences



(a) Number of unique connections each month (b) Number of clusters each month

Figure 6.9: Activity in the Enron email network from April 2001 to March 2002. (a): the number of unique connections per month. (b): the median value of the posterior distribution of K for each month, obtained using BDMCMC to estimate the LPCM. Values range from a minimum of $K = 2$ to maximum $K = 5$.



(a) September 2000 (b) October 2001

Figure 6.10: Posterior distribution over K for two months of activity of the Enron network.

Month	1	2	3	4	5
Sept	77	46	25	0	0
Oct	77	47	24	0	0
Nov	57	53	39	0	0
Dec	50	41	34	23	0
Jan	86	40	21	0	0
Feb	49	38	34	27	0
Mar	44	40	38	27	0
Apr	47	42	38	21	0
May	72	41	35	0	0
Jun	70	50	27	0	0
Jul	53	41	28	26	0
Aug	62	36	35	15	0
Sept	47	36	35	29	0
Oct	39	37	30	28	15
Nov	44	29	28	28	18
Dec	49	39	39	21	0
Jan	53	40	31	23	0
Feb	82	50	16	0	0
Mar	100	48	0	0	0

Table 6.1: Mean of the posterior distributions of the mixing proportions of the clusters, for the Enron dataset, September 2000 - March 2002

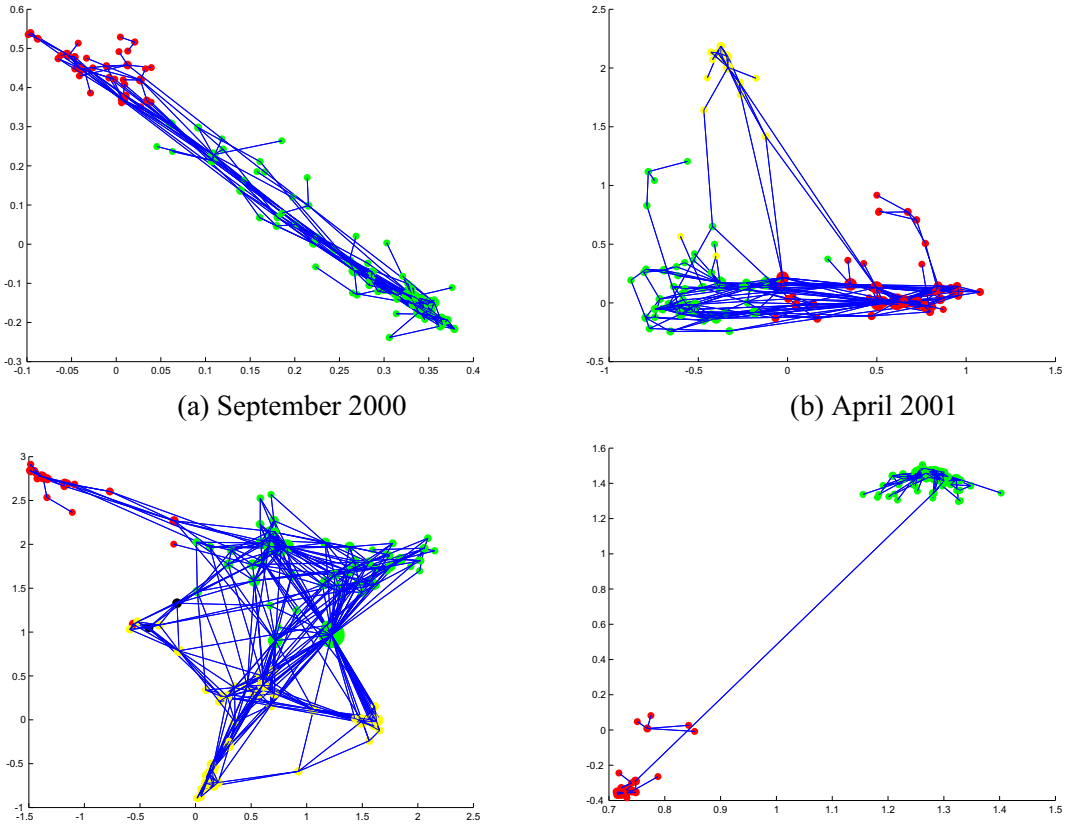
are never large, generally going from $K = 4$ to $K = 3$ or 5. Example posterior distributions over K for September 2000 and October 2001 show that there is a broad distribution over K that is peaked but still places significant probability on the values of K mentioned. Table 6.1 shows the number of employees in each cluster and it can be seen that there is often agreement between months with the same K where the community structure is consistent for example the period of February through April shows the network is made up of 4 clusters of roughly the same size.

To conclude our analysis we compared the series of embeddings obtained from the LPCM, four examples of which are shown in Figure 6.11, using the CCA method described in Section 6.2 and outlined in Figure 6.3. The four elements of the test statistic are plotted in Figure 6.12 where again 95% confidence intervals are chosen.

The plots show some volatility but the only 'real' changepoint occurs when comparing the activity of January with February. At this time Enron has been declared bankrupt and suffered the resignation of the CEO, email traffic from this period is low, see Figure 6.9a, and what is observed is effectively a breakdown of the network. Table 6.1 reveals that February marks a shift in both K and the relative sizes of the clusters from the generally similar network structure observed from September to January.

It is perhaps surprising that during a period of so many significant events more changes in the activity are not detected. The fact that we are taking snapshots of the network on a monthly basis may not be a fine enough scale to detect the effect of these events. The counter argument however is that this broader snapshot reduces the probability of false positives and also increases the amount of activity observed rather than comparing numerous empty nodes.

Considering these results in conjunction with those obtained from the experiments in Section 6.4.1 it may be that the types of changes that are occurring are not sufficient to produce a changepoint detectable by this method. Unlike the VAST data where there is a complete shift in the structure of the network due to the main actors in the network changing phones, it is hard to imagine such a change naturally occurring in an email network.



(a) September 2000 (b) April 2001

(c) October 2001 (d) February 2002

Figure 6.11: Embeddings of the activity in the Enron email network for four months. The posterior means of the positions, z, are plotted and coloured according to cluster membership.

6.5 Comparison with Existing Approaches

The method presented in this chapter is a novel contribution to the network change detection literature that makes use of a well studied model in a novel way. We have demonstrated empirically the properties and effectiveness of this approach in terms of real data analysis but it is also useful to consider how the method fits with the existing approaches outlined in Section 6.1. The methods that are most related to our approach are those based on the use of network features. The egonet approach [4, 3] closely resembles our combined LPCM-CCA approach making use of changes in correlation between calculated features to give an indication of the occurrence of a change in a network. The key difference however is in the consistent use of latent variable framework to transform the network and also when

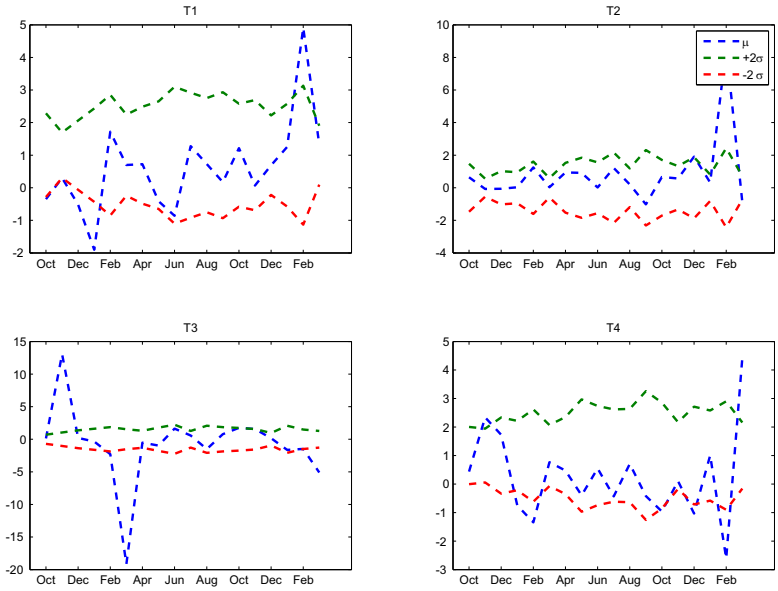


Figure 6.12: Test statistics for the Enron Email network from September 2000 to March 2002. T1, T2 and T4 all show a change occurring upon the comparison of January 2002 with February 2002.

calculating correlations. This allows us to define a test statistic and associated probabilities of a change based on observed correlations rather than their approach of monitoring the pearson's correlation coefficient associated. The use of a latent variable model approach to extract a multivariate representation of a network also contrasts between the arbitrary selection of features as in both the egonet approach and that advocated by [87, 86]. This makes explicit assumptions about the existence of an embedding space for the network and that the observed network behaviour can be explained by a model of this type. This assumption lends more power to the overall change detection method and in the case of comparing networks generated by different models we have demonstrated it's ability to detect changes. As stated previously the drawback of monitoring specific features is that changes may occur that do not manifest as changes in these specific features and lead to undetected events. However it must also be pointed out that we have defined a model and that for some networks such a model may be inappropriate and this method invalid. In such situations, for example monitoring changes in random graphs these feature based ap-

proaches may be more appropriate. The methods based on graph spectra [136] are another alternative approach that has proved popular however they tend to be more suited to the task of anomaly detection and fully connected graphs. Given that many efficient methods for computing the eigen-decomposition of large matrices exist and are a constant area of research this approach may be more suited to large graphs than that outlined here however questions do remain over which form of the graph Laplacian to use and this can produce different results based on different choices. In contrast the graph matching approach proposed in [18, 27] is extremely computationally intensive in comparison to our method. This approach coming from the structural pattern recognition branch of research [19] is most different in philosophy to the combined embedding-CCA approach outlined here, however it can be useful for identifying particularly unusual subgraphs in a network as opposed to our approach which simply states whether a global change has occurred. This makes the subgraph matching approach more suited to fraud detection based applications looking to detect unusual cliques or subgraphs within an otherwise healthy network, in this setting our approach would not be suitable and in contrast sits well with applications where we are expecting large shifts in network structure such as epilepsy seizure onset detection or in applications where the LPCM is appropriate for example monitoring changes in social networks.

6.6 Conclusion

Throughout this thesis we have made use of latent variables as a convenient means of inference and also as the target of inference themselves. In this chapter we utilised the latent positions associated with the LPCM as a representation of a network rather than just a convenient means of taking into account dependencies in the network as previously. This allowed us to deploy CCA to compare two such representations of a network. The advantages of this approach are that it is less computationally demanding than iterative graph searching methods and more rigorous than feature selection based methods to detect changes in a network. A latent variable model phrasing of CCA allows us to define probability distributions over the associated transformation matrices to create a test statistic with confidence intervals allowing us to say how likely it is that a change has occurred. The method was

tested on synthetic and real data and from the results obtained we learned that the types of changes detected tend to be of a large scale network breakdown type and as the size of the network increases the number of nodes disturbed must be increased. These characteristics may make the method attractive for network monitoring systems as the types of changes that we can pick up may be precursors to a breakdown of the network. This behaviour is a consequence of the assumptions made as part of the test statistic procedure, our requirement for at least three elements to individually flag a change. Different assumptions may alter the behaviour of the approach, but under the assumptions here the method may be more appropriate for applications such as epilepsy seizure onset detection that produce a large shift in the overall network rather than more localised changes such as detecting fraud in a network of financial transactions where the majority of interactions will be healthy and the change occurs in a significantly smaller portion of the network.

Chapter 7

Discussion and Future Work

We now conclude with a discussion of some of the key findings and potential avenues for future work.

The bulk of this thesis has been concerned with methods for analysing network data, however to begin with we studied the MFA model and Bayesian methods for simultaneously inferring both the value of K, the number of mixtures in the model, and q_k the number of factors in each factor analyser. The complexity of the model estimation task in this setting called for the use of sophisticated approximation techniques with the ability to compare model spaces of varying dimensions. We studied two such methods and provided an empirical comparison of their performance in a range of different settings in a simulation study with synthetic data. The study is related to very recent work by [21] but extended from a regression setting to unsupervised analysis. The key finding here was that given sufficient amounts of data, estimation by variational approximation can give comparable performance to MCMC based methods, measured in terms of correct estimation of K and q_k, but in compute times that are orders of magnitude faster. This also highlights the need for care when deploying variational methods to tasks where the data is sparse.

Following the simulation study the MFA model was applied to the IMGSAC ADI-R data set, containing behavioural information on a cohort of Autistic subjects from the IMGSAC study. This work represents the first use of such sophisticated methods to anal-

yse ADI-R data and the inference over both K and q_k revealed both the presence of sub-populations in a carefully selected sample and the differences in latent structure between these sub-groups. These variations in latent structure when combined with our interpretation of a latent 'severity factor' suggest that the different groups can be characterised as having different levels of impairments in the four domains of autism (social interaction, verbal and non-verbal communication skills and repetitive and restricted behaviour). However the different factor structure highlights that we are not just seeing reduced or increased levels of the same traits but that the characteristics of the between cluster behaviour are markedly different. This implies that the condition manifests differently between clusters. This is important information in the context of further analysis, large sample studies aimed at pinpointing autism phenotypes have been frustrated by the heterogeneous structure of the data. The method we have demonstrated provides an unsupervised approach for clustering the data into homogeneous sub-populations, further analysis for example of the genetic structure of patients could then be focused on these individual groups as more coherent samples. Additionally correct identification of the group a patient belongs to could allow for more appropriate treatment to meet their specific needs.

Chapter 4 introduced the LPCM which featured for the remainder of the thesis. The model allows for Bayesian model based embedding of a network in Euclidean space. The original MCMC model estimation method was extended to incorporate the Birth-Death mechanism from Chapter 3 which allowed for inference over K and was the first instance of the application of stochastic model selection methods to the model. This represents an improvement over the original approach to model selection which estimated the model for different values of K and compared pairs of models using Bayes factors. Indeed it was remarked as part of the discussion on the original RSS read paper [58] that the stochastic model selection approach is a superior method of tackling this problem. The full framework from Chapter 3, combining inference over K with that of the embedding dimension, did not translate well into the network setting and this is something to be addressed further in future work. Another issue with the analysis of network data is the massive computational burden associated with scaling up to larger networks. Informed by our findings from the simulation study we developed an approach to estimating the LPCM based on varia-

tional approximation. Previous work on this had been forced to result to a large amount of numerical optimization routines to estimate model parameters however an alternative phrasing of the model allowed us to develop a fully probabilistic solution.

The embedding produced by the LPCM was originally used simply to visualise the network and to determine the community structure. In Chapter 6 we make use of this representation of the network as a convenient means of deploying standard multivariate analysis techniques in a network setting. We developed a change detection method based on comparing representations of networks produced by the LPCM using CCA. This is the first combination of these techniques and comparable methods from the current state of the art rely on feature extraction as a means of detecting changes in a network. We see this approach as a less principled method than embedding the network as arbitrary choices of feature can be made. The method is demonstrated on some synthetic experiments and we make an effort to characterise the type of changes that can be detected. The behaviour obtained indicates that as the size of the network increases the magnitude of the change required also increases and an appropriate application may be the monitoring of networks for breakdown as these seem to be the types of changes that can be detected. We followed our simulation study with the application of our method to some data sets with mixed results. The method was able to detect the induced change in the VAST network however in the Enron data the only major change point detected occurs at the very end of the period of analysis and events such as the resignation of the CEO did not have a noticeable effect. This again implies the most appropriate deployment of our method may be in the monitoring for serious network breakdown.

7.1 Future Work

There are a number of areas for further exploration deriving from the work carried out thus far. Chief amongst these is the question of the embedding dimension and its implications but there is also potential for further analysis of the ADI-R data and development of more sophisticated methods for the test statistic.

7.1.1 ADI-R Data

The ADI-R is a structured interview consisting of approximately 100 questions scored as integers from $0-3$. These are transformed using an algorithm to obtain the 12 sub-domain scores which are then combined to obtain the scores in the four domains of autism to allow diagnosis to be made. The analysis carried out has been restricted to the domain and sub-domain levels for which we have data. It would however also be of interest to study the raw data itself and learn what kind of structure can be detected at this level. The MFA is designed for continuous data and would be unsuitable for this task however the logistic function used in the LPCM can be adapted to multiclass settings. Combined with the mixture framework again as in the LPCM this would allow us to explore the data at the item level for cluster structure.

The analysis of complementary gene expression data, informed by the results of our ADI-R analysis is another avenue of potential interest. Using the ADI-R to separate a sample into more homogeneous groups may enhance the strength of the subsequent analysis and allow for the detection of autism phenotypes linked to particular classes of subjects.

7.1.2 Embedding Dimension and Model Fit

When working with the LPCM we have generally considered the size of the embedding dimension to be fixed with $q = 2$. Section 4.2 made reference to the idea of a characteristic dimension for a network and that different graphs inhabit spaces of different size. For example the ring graph can be perfectly embedded in a 2-D space for all n, while the star graph requires $q = (n-1)/2$.

In Section 3.1 we stressed the advantages of joint estimation of the number of factors q_k and K in the MFA model and many of those arguments also apply here to reinforce the importance of the embedding dimension. Correct estimation of the embedding dimension could potentially have an effect on the number of clusters detected in a network and ideally we would want to infer it in a Bayesian fashion from the data. The dimension may be of interest in itself and also provides a further means of detecting changes in network structure which could be incorporated into a three tiered approach:

- Infer q for networks.

- If q is the same compare the community structure of the graphs by comparing K.
- If q and K are the same compare with CCA.

We can try to reason about the effect of changing embedding dimension in the LPCM by considering the relationship between distance and dimension. The LPCM computes the likelihood of a link between two nodes based on the distance between them in latent space. Increasing the dimension of the latent space can be seen as reducing the overall tendency to form links in the network. This can be understood by considering a hub or star network. In this network a number of nodes connect to a single hub node. In low dimensions there are fewer ways to arrange this network without positioning nodes that are disconnected closer to each other than the hub node. This would break the distance rule of the Latent Position model, that nodes closer together in latent space are more likely to have a link. The model assumptions could only hold if the network became more connected and if the embedding positions were re-used to generate the network it would be more heavily connected. Increasing the dimension however increases the number of positions available for a node to inhabit while still maintaining the same distance. In the model β, see Equation 4.4, performs a similar role modulating the overall preference for forming links in the network. In some preliminary experiments we conducted to estimate the embedding dimension, by integrating over all possible dimensions, we found that there was an interplay between β and the embedding dimension. We found that β acted as a counterweight adapting between different states of dimension making it impossible to correctly choose between them. Further research is needed to fully understand and resolve this problem.

7.1.3 Estimation of the LPCM by Variational Approximations

There are a number of ways the estimation of the LPCM by variational approximations presented in Chapter 5 could be extended. Incorporation of inference over hyperparameters would be an obvious means of making the model more flexible. As remarked in the chapter the use of the von Mises distribution restricts the embedding dimension to $q = 2$. It may be possible to generalise q to other dimensions by using the Fisher-Bingham distribution which a generalisation of the von Mises distribution to higher dimensions. However the

relationship between the von Mises distribution and the bivariate Gaussian distribution was a key component in the derivation of the variational posteriors and whether this relationship would still hold remains unclear. Finally model selection over the value of K could be incorporated by following the same approach used in Chapter 3 for the MFA model estimated by variational approximations. This would allow unsupervised community detection in a network as demonstrated in Chapters 4 and 6 using the birth-death MCMC approach but with the added advantage obtained by the superior speed of the variational algorithm. Clearly there is scope for additional work here however much is dependent upon a solution to the embedding dimension problem.

7.1.4 Test Statistic

In Chapter 6 we have a matrix valued test statistic to detect change in a network, each value has a distribution and we compare each individual element with the corresponding element of an identity matrix to determine whether a change has occurred. However if we are to allow the embedding dimension to grow then the size of this matrix will also increase making our current scheme impractical, a method based on weighted combinations of the elements would be an improvement on this and is an area that could be considered for further study. Methods based on random matrix theory could also be considered as an alternative to evaluating each element individually.

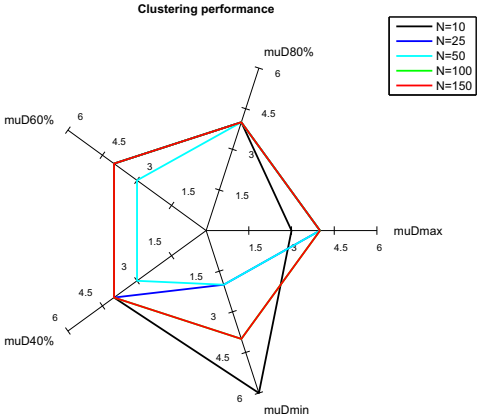
Appendix A

Simulation Study Results

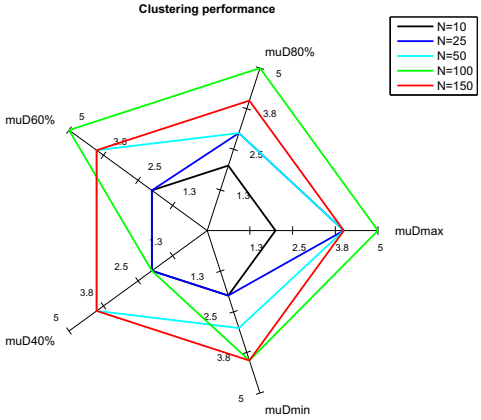
In Chapter 3 we conducted an empirical comparison of the MCMC and variational approaches to estimating the MFA model is a range of different settings. The results obtained for the experiments with $p = \{20, 30\}$ are provided here with spider plots used to summarise the performance obtained estimating K as the difficulty of the clustering problem is increased for the number of points per cluster $n_k \in \{10, 25, 50, 150\}$. The results from estimating q_k are summarised by plotting the string edit distance between the results and the true $\boldsymbol{q} = [1, 2, 3, 4]$. Results are shown for data drawn from the model in Section A.1 and for data that features a departure from the model assumptions in Section A.2, the departure being additive noise from a Uniform rather than Gaussian distribution.

A.1 Results for Data from the Model

The results obtained for $p = \{20, 30\}$ using synthetic data generated from the model.

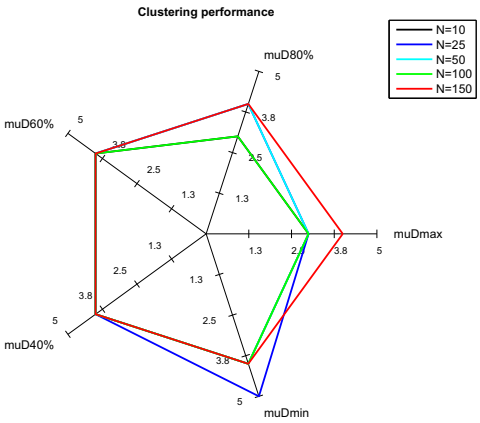


(a) MCMC spider plot for K, $p = 20$.

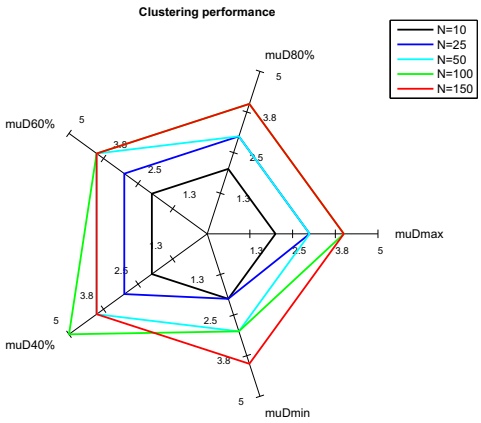


(b) VB spider plot for K, $p = 20$

Figure A.1: Results obtained from estimation of K varying the number of points per cluster and the distance between clusters, using the MCMC (a) and variational (b) methods respectively.

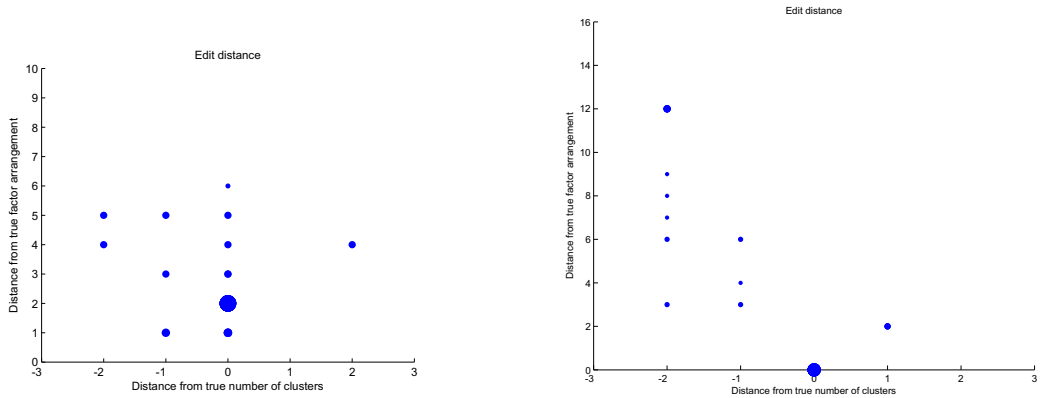


(a) MCMC spider plot for K, $p = 30$.



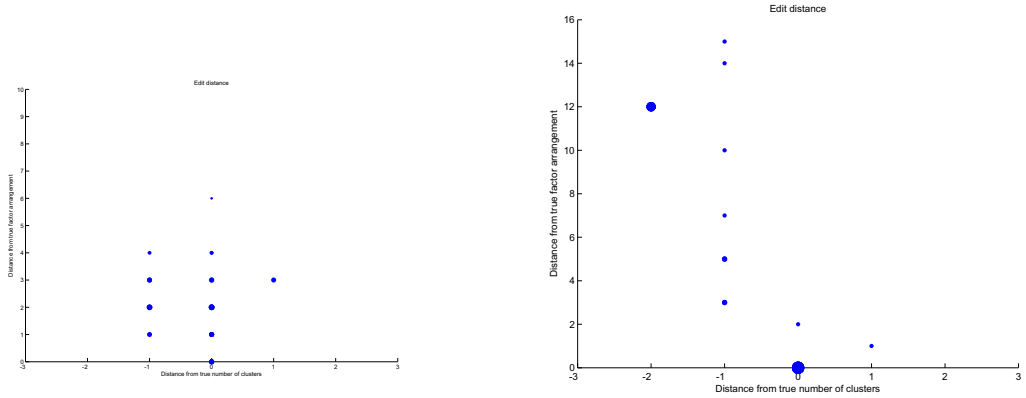
(b) VB spider plot for K, $p = 30$

Figure A.2: Results obtained from estimation of K varying the number of points per cluster and the distance between clusters, using the MCMC (a) and variational (b) methods respectively.



(a) MCMC factors, $p = 20$ (b) VB factors, $p = 20$

Figure A.3: Results obtained for the comparison of the string edit distance between the true $\boldsymbol{q}$ and that estimated by the MCMC (a) and variational (b) methods. Each plot consists of 25 results and the size of the data point reflects multiple occurrences of the same integer value. The x-axis shows the distance from true K with the origin representing correct estimation of both $\boldsymbol{q}$ and K.

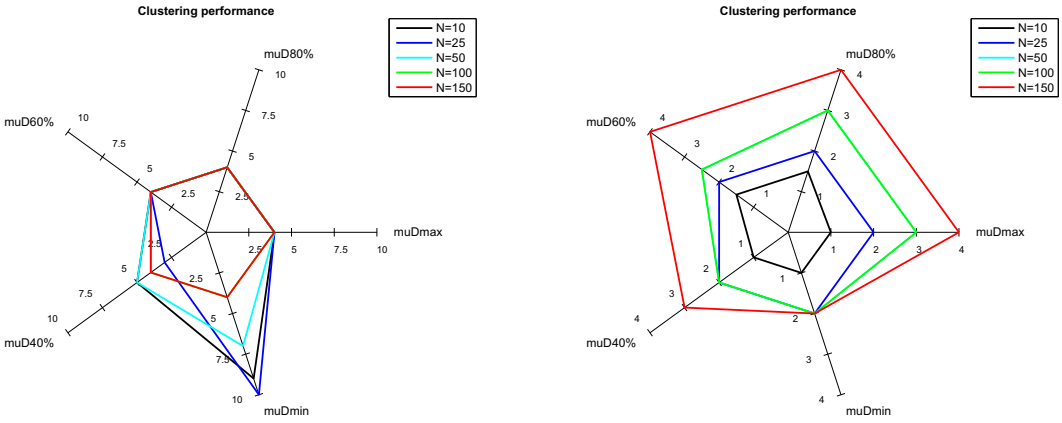


(a) MCMC factors, $p = 30$ (b) VB factors, $p = 30$

Figure A.4: The results obtained for the comparison of the string edit distance between the true $\boldsymbol{q}$ and that estimated by the MCMC (a) and variational (b) methods. Each plot consists of 25 results and the size of the data point reflects multiple occurrences of the same integer value. The x-axis shows the distance from true K with the origin representing correct estimation of both $\boldsymbol{q}$ and K.

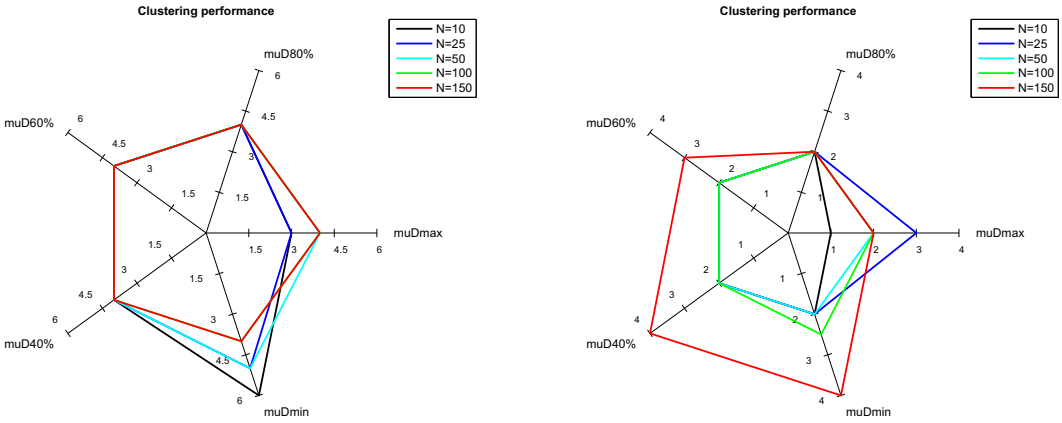
A.2 Results for Data with Uniform Noise

The results obtained for $p = \{20, 30\}$ using synthetic data that does not obey the model, is subject to random noise from a Uniform distribution.



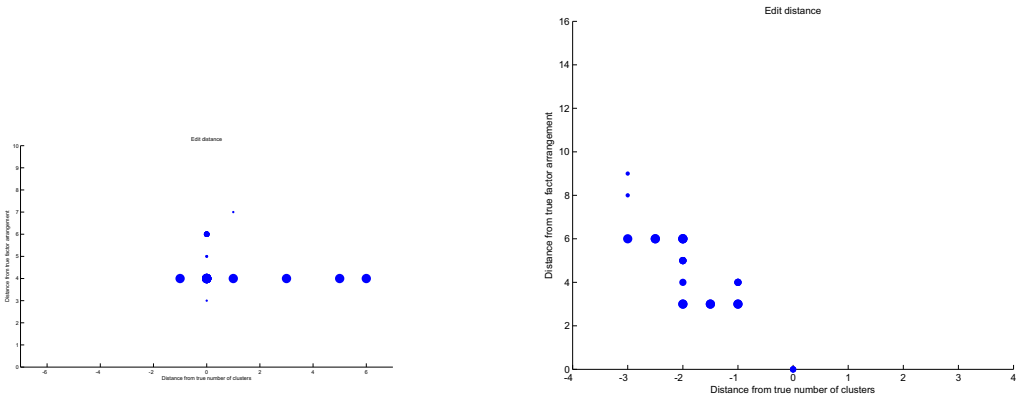
(a) MCMC spider plot, $p = 20$, uniform noise. (b) VB spider plot, $p = 20$, uniform noise.

Figure A.5: Results obtained for estimation of K when data deviates from model assumptions i.e. is subject to uniform noise by MCMC (a) and variational (b) methods.



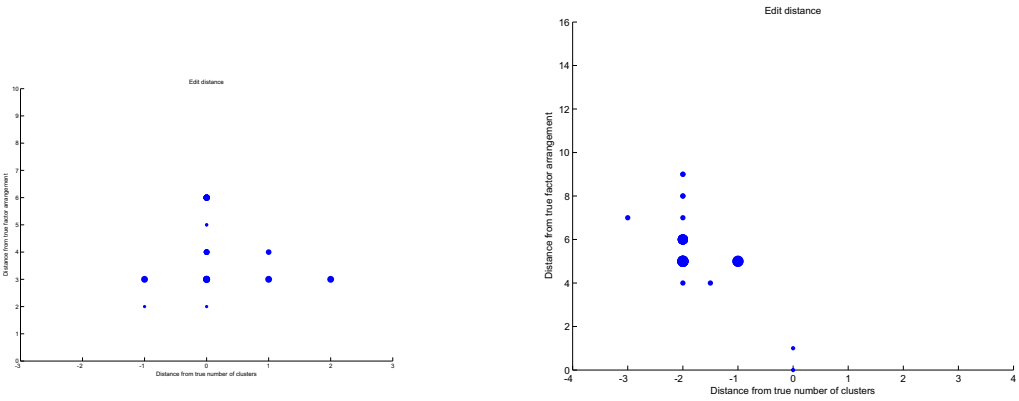
(a) MCMC spider plot, $p = 30$, uniform noise. (b) VB spider plot, $p = 30$, uniform noise.

Figure A.6: Results obtained for estimation of K when data deviates from model assumptions i.e. is subject to uniform noise by MCMC (a) and variational (b) methods.



(a) MCMC factors, $p = 20$, uniform noise. (b) VB factors, $p = 20$, uniform noise.

Figure A.7: Results obtained for estimation of $\boldsymbol{q}$ using MCMC (a) and variational (b) methods when data deviates from model assumptions i.e. is subject to uniform noise with $p = 20$. Each plot consists of 25 results and the size of the data point reflects multiple occurrences of the same integer value. The x-axis shows the distance from true K with the origin representing correct estimation of both $\boldsymbol{q}$ and K.



(a) MCMC factors, $p = 30$, uniform noise. (b) VB factors, $p = 30$, uniform noise.

Figure A.8: Results obtained for estimation of $\boldsymbol{q}$ using MCMC (a) and variational (b) methods when data deviates from model assumptions i.e. is subject to uniform noise with $p = 30$. Each plot consists of 25 results and the size of the data point reflects multiple occurrences of the same integer value. The x-axis shows the distance from true K with the origin representing correct estimation of both $\boldsymbol{q}$ and K..

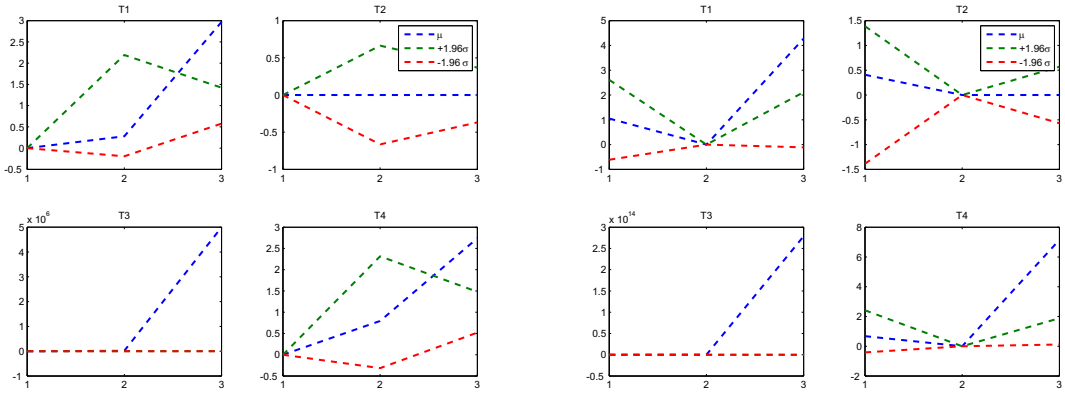
Appendix B

Change Magnitude Experiment Results

The results of the experiments conducted in Chapter 6 to assess the magnitude of the change that can be detected using the combined embedding-CCA method are provided here. The general trends observed in the experiment are discussed in Section 6.4.1 but here the raw results are provided for completeness. The test statistics for the comparison of networks subjected to increasing levels of change against an original network generated from the latent position model are shown for different network sizes $n = \{10, 25, 50, 100\}$. Section B.1 shows the results obtained for the random rewiring experiment, and Section B.2 provides the results for the experiment in which nodes are randomly rewired but the degree sequence of the graph preserved.

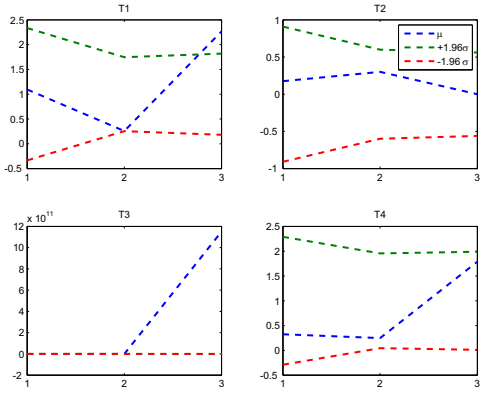
B.1 Random Rewiring Experiment

In this experiment nodes were randomly selected and rewired starting with a single node and increasing in steps of 20% to 80% of the total network. These networks were embedded using the latent position model and compared with the original network using CCA and the test statistics obtained by comparison of the transformation matrices are shown.

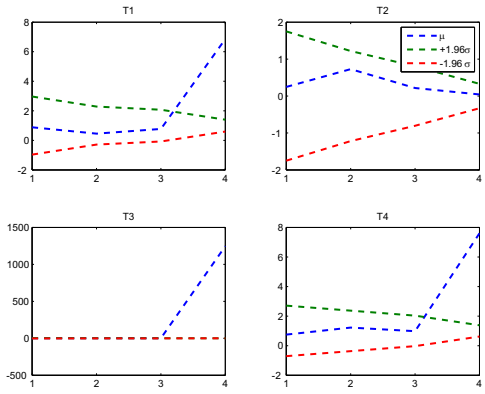


(a) Test Statistic $n = 10$ random rewiring

(b) Test Statistic $n = 25$ random rewiring



(c) Test Statistic $n = 50$ random rewiring

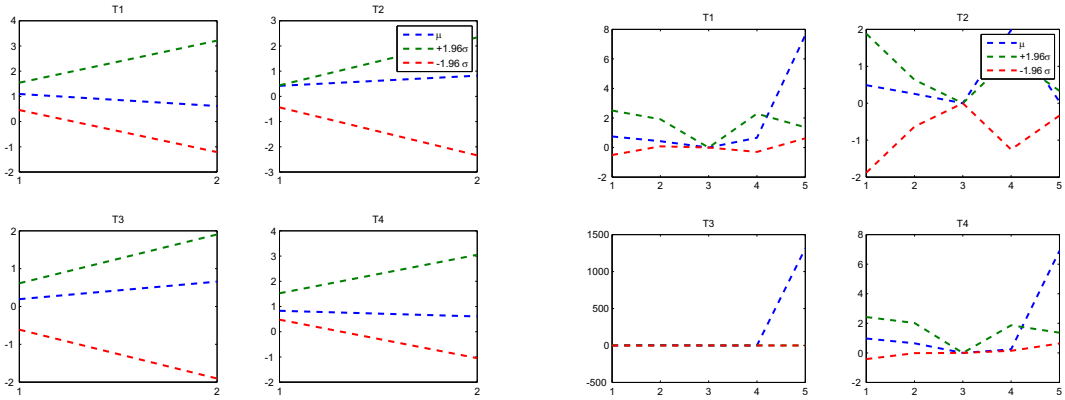


(d) Test Statistic $n = 100$ random rewiring

Figure B.1: Results for the random rewiring change detection experiment. The test statistics obtained from the comparison of an original network to networks where a number of the nodes have been randomly rewired. The number of nodes rewired is increased in steps of 20% from a single node to 80% off all nodes for $n \in \{10, 25, 50, 100\}$.

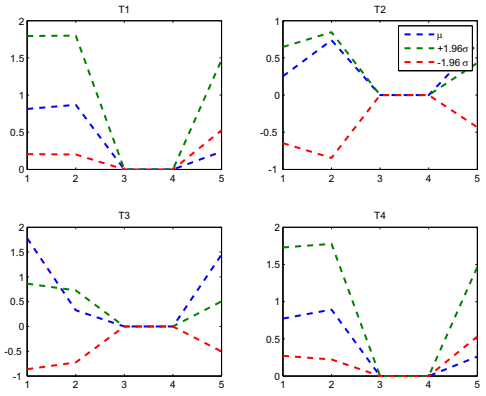
B.2 Degree Sequence Rewiring Experiment

In this experiment nodes were randomly selected and rewired but the degree sequence of the graph preserved. The number of nodes rewired starts from a single node and increases in steps of 20% to 80% of the total network. These networks were embedded using the latent position model and compared with the original network using CCA and the test statistics obtained by comparison of the transformation matrices are shown

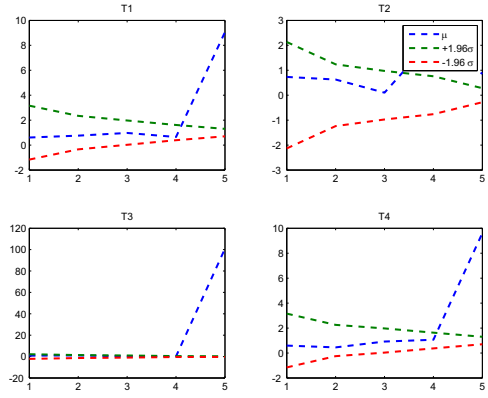


(a) Test Statistic $n = 10$

(b) Test Statistic $n = 25$



(c) Test Statistic $n = 50$



(d) Test Statistic $n = 100$

Figure B.2: Results for the random rewiring change detection experiment. The test statistics obtained from the comparison of an original network to networks where a number of the nodes have been rewired but the degree sequence of the network is unchanged. The number of nodes rewired is increased in steps of 20% from a single node to 80% off all nodes for $n \in \{10, 25, 50, 100\}$.

References

[1] Omolabake Adenle and William Fitzgerald. Bayesian Model Selection for Independent Factor Analysis. *IEEE Information Theory Workshop*, 1:337–341, 2006. 54

[2] Hirotugu Akaike. A New Look at Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19:716–723, 1974. 36

[3] Leman Akoglu and Christos Faloutsos. Event Detection in Time Series of Mobile Communication Graphs. *Proceedings of Army Science Conference*, pages 1–8, 2010. 137, 157

[4] Leman Akoglu, Mary McGlohon, and Christos Faloutsos. OddBall: Spotting Anomalies in Weighted Graphs. *Proceedings of the 14th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining*, 2:410–421, 2010. 136, 137, 157

[5] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael Jordan. An Introduction to MCMC for Machine Learning. *Machine Learning*, 50:5–45, 2003. 39, 40, 41

[6] Hagai Attias. Inferring Parameters and Structure of Latent Variable Models by Variational Bayes. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 21–30, 1999. 39

[7] Francis Bach and Michael Jordan. A Probabilistic Interpretation of Canonical Correlation Analysis. *Technical Report 688. University of California, Berkeley*, 2005. 30, 31, 139

[8] Albert-Laszlo Barabási and Réka Albert. Emergence of Scaling in Random Networks. *Science*, 286:509–512, 1999. 95

[9] David Bartholomew. *Latent Variable Models and Factor Analysis*. Oxford University Press, 1987. 24, 26, 29

[10] Matthew Beal. *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, University College London, 2003. 47, 63, 64, 66, 134

[11] Matthew Beal and Zoubin Ghahramani. The Variational Bayesian EM Algorithm for Incomplete Data: with Application to Scoring Graphical Model Structures. *Bayesian Statistics*, 7:000–000, 2003. 43

[12] Ginestra Bianconi. The Entropy of Randomized Network Ensembles. *Europhysics Letters*, 81(2):1–6, 2008. 98

[13] Christopher Bishop. *Learning in Graphical Models*, chapter Latent Variable Models, page 371403. MIT Press, 1999. 24, 43

[14] Christopher Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. 24, 28, 32, 34, 35, 36, 37, 38, 39, 41, 42, 44, 50, 83, 111, 116, 118, 123, 129, 130

[15] Magnus Borga. Canonical Correlation: a Tutorial. *Technical Report*, at http://people.imt.liu.se/∼magnus/cca/tutorial/, 2001. 29

[16] Yacine Bouzida, Frederic Cuppens, Nora Cuppens-Boulahia, and Sylvain Gombault. Efficient Intrusion Detection using Principal Component Analysis. *Proceedings of the 3e'me Conference sur la Securite et Architectures Reseaux*, pages 1–12, 2004. 28

[17] Norman Breslow. Statistics in Epidemiology: The Case-Control Study. *Journal of the American Statistical Association*, 91:14–28, 1996. 103

[18] Horst Bunke, Peter Dickinson, Andreas Humm, Christophe Irniger, and Miro Kraetzl. Computer Network Monitoring and Abnormal Event Detection Using

Graph Matching and Multidimensional Scaling. *International Conference on Data Mining*, 4065:576–590, 2006. 137, 138, 139, 159

[19] Horst Bunke and Kasper Riesen. Towards the Unification of Structural and Statistical Pattern Recognition. *Pattern Recognition Letters*, 33:811–825, 2011. 109, 159

[20] Olivier Cappe, Christian Robert, and Tobias Ryden. Reversible Jump, Birth-and-Death, and more General Continuous Time MCMC Samplers. *Journal of the Royal Statistical Society Series B - Statistical Methodology*, 65:679–700, 2003. 54

[21] Peter Carbonetto and Matthew Stephens. Scalable Variational Inference for Bayesian Variable Selection in Regression and its Accuracy in Genetic Association Studies. *Bayesian Analysis*, 7:73–108, 2012. 15, 17, 65, 66, 71, 161

[22] Gilles Celeux, Merrilee Hurn, and Christian P. Robert. Computational and Inferential Difficulties with Mixture Posterior Distributions. *Journal of American Statistical Association*, 95:957–970, 2000. 61

[23] John Chambers, William Cleveland, Beat Kleiner, and Paul Tukey. *Graphical Methods for Data Analysis*. Wadsworth, 1983. 67

[24] Anurat Chapanond, Mukkai Kirishnamoorthy, and Bulent Yener. Graph Theoretic and Spectral Analysis of Enron Email Data. *Proceedings of Workshop on Link Analysis. Counterterrorism and Security, SIAM International Conference on Data Mining*, pages 15–22, 2005. 152

[25] Reuven Cohen and Shlomo Havlin. *Complex Networks Structure, Robustness and Function*. Cambridge, 2010. 90, 95

[26] John Constantino, Christian Gruber, Sandra Davis, Stephanie Hayes, Natalie Passante, and Thomas Przybeck. The Factor Structure of Autistic Traits. *Child Psychology and Psychiatry*, 45:4:719–726, 2004. 74, 75

[27] Donatello Conte, Pasquale Foggia, Carlo Sansone, and Mario Vento. Thirty Years of Graph Matching in Pattern Recognition. *International Journal of Pattern Recognition and Artic̄ial Intelligence*, 18:265–298, 2004. 18, 138, 159

[28] Trevor Cox and Michael Cox. *Multidimensional Scaling*. Chapman and Hall, 2000. 27, 98, 102

[29] Carl Craig. On the Frequency Function of xy. *Annals of Mathematical Statistics*, 7:1–15, 1936. 141

[30] Michael Cuccaro, Yujan Shao, Janet Grubber, Michael Slifer, Chantelle. Wolpert, and Shannon Donnelly. Factor Analysis of Restricted and Repetitive Behaviors in Autism using the Autism Diagnostic Interview - R. *Child Psychiatry and Human Development*, 34:3–17, 2003. 74, 75

[31] John Diebolt and Christian P. Robert. Estimation of Finite Mixture Distributions Through Bayesian Sampling. *Journal of the Royal Statistical Society. Series B*, 56:363–375, 1994. 61

[32] Jana Diesner, Terrill Frantz, and Kathleen Carley. Communication Networks from the Enron Email Corpus: 'It's Always About the People. Enron is no Different'. *Institure for Software Research*, 46:201–228, 2006. 151

[33] Clare Dover and Ann Le Couteur. How to Diagnose Autism. *Archives of Disease in Childhood*, 92:540–545, 2007. 73

[34] William Eberle and Lawrence Holder. Discovering Structural Anomalies in Graph-Based Data. *IEEE Conference on Data Mining - Workshops*, pages 393–398, 2007. 136, 137

[35] Alan Edelman and Raj Rao. Random Martix Theory. *Acta Numerica*, 24:1–65, 2005. 140

[36] Bradley Efron. Bootstrap Methods: Another Look at the Jackknife. *Annals of Statistics*, 7:1–26, 1979. 36

[37] Paul Erdős, Frank Harary, and William Tutte. On the Dimension of a Graph. *Mathematika*, 12:118–122, 1965. 98

[38] Paul Erdős and Alfred Rényi. On Random Graphs. *Publicationes Mathematicae*, 6:290–297, 1959. 91

[39] Paul Erdős and Alfred Rényi. On the Evolution of Random Graphs. *Publications of the Hungarian Academy of Sciences*, 5:17–61, 1960. 91

[40] Paul Erdős and Alfred Rényi. On the Strength of Connectedness of a Random Graph. *Acta Mathematica Scientia Hungary*, 12:261–267, 1961. 91

[41] Leonhard Euler. Solutio Problematis ad Geometriam Situs Pertinentis. *Commentarii academiae scientiarum Petropolitanae*, 8:128–140, 1736. 90

[42] Brian Everitt. *An Introduction to Latent Variable Models*. Chapman and Hall, 1984. 26

[43] Katherine Faust. Comparison of Methods for Positional Analysis: Structural and General Equivalence. *Social Networks*, 19:313–341, 1988. 98

[44] Stephen Fienberg. A Brief History of Statistical Models for Network Analysis and Open Challenges. *Journal of Computational and Graphical Statistics*, 21:825–839, 2012. 91, 92, 108, 145, 148

[45] Ronald Fisher. *Statistical Methods for Research Workers*. Oliver and Boyd, Edinburgh, 1925. 23

[46] Ronald Fisher. *The Design of Experiments*. Oliver and Boyd Edinburgh, 1935. 23

[47] Ronald Fisher. Dispersion on a Sphere. *Proceedings of the Royal Society of London, Series A*, 217:295–305, 1953. 117

[48] Ernest Fokoue and Mike Titterington. Stochastic Model Selection for Bayesian Mixtures of Factor Analysers. *Technical Report, Department of Statistics, University of Glasgow.*, 005:1–10, 2000. 49, 54, 59, 62

[49] Ernest Fokoue and Mike Titterington. Mixtures of Factor Analysers. Bayesian Estimation and Inference by Stochastic Simulation. *Machine Learning*, 50:73–94, 2003. 27, 33, 34, 48, 50, 53, 54, 58

[50] Thomas Frazier, Eric Youngstrom, Cynthia Kubu, Leslie Sinclai, and Ali Rezai. Exploaratory and Confirmatory Factor Analysis of the Autism Diagnostic Interview - Revised. *Journal of Autism and Developmental Disorders*, 38:474–480, 2007. 74, 75, 77

[51] Stuart Geman and Donald Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984. 42

[52] Zoubin Ghahramani and Matthew Beal. Variational Inference for Bayesian Mixtures of Factor Analysers. *Advances in Neural Information Processing*, 2000. 63, 64, 65, 66

[53] Zoubin Ghahramani and Geoffrey E. Hinton. The EM Algorithm for Mixtures of Factor Analysers. *Technical Report CRG-TR-96-1, Dept. Computer Science, Univ. of Toronto*, 1:1, 1996. 26, 46

[54] Walter Gilks, Sylvia Richardson, and David Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, 1996. 39, 40, 41

[55] Anna Goldenberg, Alice Zheng, Stephen Fienberg, and Edoardo Airoldi. A Survey of Statistical Network Models. *Foundations and Trends in Machine Learning*, 2:1–117, 2009. 92

[56] Geroges Grinstein, Catherine Plaisant, Sharon Laskowski, Theresa O'Connell, Jean Scholtz, and Mark Whiting. VAST 2008 Challenge: Introducing Mini-Challenges. *IEEE Symposium on Visual Analytics Science and Technology*, 1:195–196, 2008. 107, 149, 150

[57] Dan Gusfield. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997. 67

[58] Mark Handcock and Adrian Raftery. Model-Based Clustering for Social Networks. *Journal of the Royal Statistical Society Series A*, 170(2):1–22, 2007. 16, 18, 88, 92, 99, 101, 105, 132, 133, 134, 162

[59] Keith Hastings. Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika*, 57:97–109, 1970. 41

[60] Nicholas Heard, David Weston, Kiriaki Platanioti, and David Hand. Bayesian Anomaly Detection Methods For Social Networks. *The Annals of Applied Statistics*, 4,:645–662, 2010. 107, 150

[61] Geoffrey Hinton. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, chapter Distributed Representations, pages 77–109. MIT Press, 1986. 79, 81

[62] Geoffrey Hinton, Peter Dayan, and Michael Revow. Modelling the Manifolds of Images of Handwritten Digits. *IEEE transactions on Neural Networks*, 8:65–74, 1997. 29

[63] Peter Hoff, Adrian Raftery, and Mark Handcock. Latent Space Approaches to Social Network Analysis. *Journal of the American Statistical Association*, 97:1090–1098, 2002. 16, 88, 91, 96, 98, 100, 116

[64] Harold Hotelling. Analysis of a complex of Statistical Variables into Principal Components. *Journal of Educational Psychology*, 24:417–441, 1933. 28

[65] Harold Hotelling. Relations Between Two Sets of Variates. *Biometrika*, 28(3-4):321–377, 1936. 29, 139

[66] Valerie Hu and Mara Steinberg. Novel Clustering of Items From the Autism Diagnostic Interview-Revised to Define Phenotypes within Autism Spectrum Disorders. *Autism Research*, 2:67–77, 2009. 73, 74, 75

[67] Jun Huan, Wei Wang, and Jan Prins. Efficient Mining of Frequent Subgraph in the Presence of Isomorphism. *Proceedings of the 2003 International Conference on Data Mining*, pages 549–552, 2003. 18

[68] Tsuyoshi Ide and Hisashi Kashima. Eigenspace-based Anomaly Detection in Computer Systems. *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 1:440–449, 2004. 137

[69] Tommi Jaakkola and Michael Jordan. Bayesian Parameter Estimation via Variational Methods. *Statistics and Computing*, 10:25–37, 2000. 111, 112

[70] Ajay Jasra, Christopher Holmes, and David Stephens. Markov Chain Monte Carlo Methods and the Label Switching Problem in Bayesian Mixture Modelling. *Statistical Science*, 20:50–67, 2005. 61

[71] Ian Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986. 28

[72] Robert Kass and Adrian Raftery. Bayes Factors. *Journal of the American Statistical Association*, 90:773–795, 1995. 37

[73] David Kendall. A Suvey of the Statistical Theory of Shape. *Statistical Science*, 4:87–99, 1989. 27

[74] John Kent. The Fisher-Bingham Distribution on the Sphere. *Journal of the Royal Statistical Society Series B*, 44:71–80, 1982. 118

[75] Michael Kirby and L.aurence Sirovich. Application of the KarhunenLoeve Procedure for the Characterization of Human Faces. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 12(1):103–107, 1990. 28

[76] Arto Klami and Samuel Klaski. Local Dependent Components. *Proceeding of the 24th International Conference on Machine Learning*, pages 425–432, 2007. 30, 32, 140

[77] Ron Kohavi. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Proceedings of the Fourteenth International Joint Conference on Artiı̄ial Intelligence*, 2(12):1137–1143, 1995. 36

[78] Eric Kolaczyk. *Statistical Analysis of Network Data*. Springer, 2009. 89, 90, 91

[79] Mark Kramer, Eric Kolaczyk, and Heidi Kirsch. Emergent Network Topology at Seizure Onset in Humans. *Epilepsy Research*, 79:173–186, 2008. 136

[80] Wojtek Krzanowski and Francis Marriott. *Kendall's Library of Statistics 2 Multivariate Analysis, Part 2*. Arnold, 1996. 24, 25, 26, 27

[81] Catherine Lord, Ann Le Couteur, and Michael Rutter. Autism Diagnostic Interview-Revised: A Revised Version of a Diagnostic Interview for Caregivers of Individuals with Possible Pervasive Developmental Disorders. *Journal of Autism and Developmental Disorders*, 24:659–685, 1994. 47, 73, 76

[82] Geoffrey Machlachlan and David Peel. *Finite Mixture Models*. Wiley, 2000. 32, 34, 35, 36, 50, 61

[83] David Mackay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003. 37, 38, 64

[84] Kanti Mardia and Peter Jupp. *Directional Statistics*. Wiley, 2000. 116, 117

[85] Frank Massey. The Kolmogorov-Smirnov Test for Goodness of Fit. *American Statistical Association*, 46:68–78, 1951. 142

[86] Ian McCulloh. *Detecting Changes in a Dynamic Social Network*. PhD thesis, School for Computer Science, Carnegie Mellon University, 2009. 138, 158

[87] Ian McCulloh and Kathleen Carley. Social Network Change Detection. *Carnegie Mellon University, School of Computer Science, Technical Report*, 2008. 18, 136, 138, 158

[88] David McFarland and Daniel Brown. Social Distance as Metric: A Systematic Introduction to Smallest Space Analysis. *In Bonds of Pluralism: the Form and Substance of Urban Social Networks*, pages 212–253, 1973. 98

[89] Geoffrey Mclachlan, David Peel, and Richard Bean. Modelling High-Dimensional Data by Mixtures of Factor Analysers. *Computational Statistics and Data Analysis*, 41:379–388, 2003. 48

[90] Nicholas Metropolis, Arianna Rosenbluth, Marshall Rosenbluth, Augusta Teller, and Edward Teller. Equations of State Calculations by Fast Computing Machine. *Jounral of Chemical Physics*, 21(6):1087–1092, 1953. 41

[91] Nicholas Metropolis and Stan Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44:335–341, 1949. 39

[92] Stanley Milgram. The Small World Problem. *Psychology Today*, 1:61–67, 1967. 91, 93

[93] Cindy Molloy, Mehdi Keddache, and Lisa J. Martin. Evidence for Linkage on 21q and 7q in a Subset of Autism Characterized by Developmental Regression. *Molecular Psychiatry*, 10:741–746, 2005. 74, 75

[94] Kevin Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012. 111

[95] Mark Newman. *Networks: An Introduction*. Oxford University Press, 2010. 97

[96] Caleb Noble and Diane Cook. Graph-Based Anomaly Detection. *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 631–636, 2003. 136, 137

[97] International Molecular Genetic Study of Autism Consortium. A Full Genome Screen for Autism with Evidence for Linkage to a Region on Chromosome 7q. *Human Molecular Genetics*, 7:571–578, 1998. 76

[98] International Molecular Genetic Study of Autism Consortium. A Genomewide Screen for Autism: Strong Evidence for Linkage to Chromosomes, 2q, 7q and 16p. *American Journal of Human Genetics*, 69:570–581, 2001. 76

[99] Aidan O'Sullivan, Niall Adams, and Iead Rezek. Canonical Correlation Analysis for Detecting Changes in Network Structure. *IEEE 12th International Conference on Data Mining*, 1:250–257, 2012. 21, 136, 142, 151, 152

[100] George Iliopoulos Panagiotis Papastamoulis. An Artificial Allocations Based Solution to the Label Switching Problem in Bayesian Analysis of Mixtures of Distributions. *Journal of Computational and Graphical Statistics*, 19(2):313–331, 2010. 61

[101] Giorgio Parisi. *Statistical Field Theory*. Addison Wesley, 1988. 43

[102] Lance Parsons, Ehtesham Haque, and Huan Liu. Evaluating Subspace Clustering Algorithms. *Proceedings of the Fourth SIAM International Conference of Data Mining, Workshop Clustering High Dimensional Data and Its Applications.*, 2004. 49

[103] Lance Parsons, Ehtesham Haque, and Huan Liu. Subspace Clustering for High Dimensional Data: A Review. *SIGKDD Explorations Newsletter*, 6:90–105, 2004. 49

[104] Karl Pearson. On Lines and Planes of Closest Fit to Systems of Points in Space. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, 2:559–572, 1901. 28

[105] Patrick Perry and Patrick Wolfe. Null Models for Network Data. *ArXiv e-prints, arXiv:1201.5871*, 2012. 96

[106] Richard Picard and Dennis Cook. Cross-Validation of Regression Models. *Journal of the American Statistical Association*, 79(387):575–583, 1984. 36

[107] Gang Qian, Shamik Sural, Yuelong Gu, and Sakti Pramanik. Similarity Between Euclidean and Cosine Angle Distance for Nearest Neighbor Queries. *Proceedings of 2004 ACM Symposium on Applied Computing*, pages 1232–1237, 2004. 116

[108] Adrian Raftery, Xiaoyue Niu, Peter Hoff, and Ka Yee Yeung. Fast Inference for the Latent Space Network Model Using a Case-Control Approximate Likelihood. *Technical Report no. 572*, pages 1–19, 2010. 89, 98, 103, 104, 108

[109] William Rand. Objective Criteria for the Evaluation of Clustering Methods. *Journal of American Statistical Association*, 66:846–850, 1971. 77

[110] Sylvia Richardson and Peter Green. On Bayesian Analysis of Mixtures with an Unknown Number of Components. *Journal of the Royal Statistical Society. Series B*, 59:731–792, 1997. 54

[111] Gareth Roberts, Andrew Gelman, and Walter Gilks. Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms. *Annals of Applied Probability*, 7(1):110–120, 1997. 42

[112] Ralph Rockafeller. *Convex Analysis*. Princeton University Press, 1972. 112

[113] Sheldon Ross. *Introduction to Probability Models*. Academic Press, 2007. 40, 41

[114] Michael Salter-Townshend and Thomas Brendan Murphy. Variational Bayesian Inference for the Latent Position Cluster Model. *Neural Information Processing Systems*, 2009. 18, 20, 111, 114, 132

[115] Samuel Sampson. *Crisis in a Cloister*. PhD thesis, Cornell University, 1969. 114, 132, 133

[116] Alberto Sanfeliu and King-Sun Fu. A Distance Measure Between Attributed Relational Graphs for Pattern Recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 13:353–362, 1983. 138

[117] Jitesh Shetty and Jafar Adibi. The Enron Email Dataset Database Schema and Brief Statistical Report. *Information Sciences Institute Technical Report, University of Southern California*, 1:1–7, 2004. 152

[118] Anders Skrondal. *Generalized Latent Variable Modeling: Multilevel, Longitudinal and Structural Equation Models.* Chapman and Hall, 2004. 24

[119] Tom Snijders. Statistical Models for Social Networks. *Annual Review of Sociology*, 37:129–151, 2011. 91

[120] Anne Snow, Luc Lecavalier, and Carrie Houts. The Structure of the Autism Diagnostic Interview - Revised: Diagnostic and Phenotypic Implications. *Child Psychology and Psychiatry*, 50:6:734–742, 2009. 73, 74, 75

[121] Charles Spearman. 'General Intelligence', Objectively Determined and Measured. *American Journal of Psychology*, 15:201–293, 1904. 23

[122] Sarah Spence, Rita Cantor, Lien Chung, Sharon Kim, Daniel Geschwind, and Maricela Alcarn. Stratification Based on Language-Related Endophenotypes in Autism: Attempt to Replicate Reported Linkage. *American Journal of Medical Genetics part B (Neuropsychiatric Genetics)*, 141B:591–598, 2006. 74, 75

[123] Donna Spiker, Linda Lotspeich, Sue Dimceli, Richard Myers, and Neil Risch. Behavioral Phenotypic Variation in Autism Multiplex Families: Evidence for a Continuous Severity Gradient. *American Journal of Medical Genetics (Neuropsychiatric Genetics)*, 114:129–136, 2002. 74, 75, 85

[124] Olaf Sporns. *Networks of the Brain*. MIT press, 2011. 89, 90, 92, 93, 95, 97

[125] Matthew Stephens. *Bayesian Methods for Mixtures of Normal Distributions*. PhD thesis, Oxford University, 1997. 53, 54

[126] Matthew Stephens. Bayesian Analysis of Mixture Models with and Unknown Number of Components - An Alternative to Reversible Jump Methods. *The Annals of Statistics*, 28:40–74, 2000. 53, 54

[127] Jimeng Sun, Spiros Papadimitriou, Philip Yu, and Christos Faloutsos. Graphscope: Parameter-free Mining of Large Time-Evolving Graphs. *Proceedings of Knowledge Discovery and Data Mining*, 2007. 152

[128] Lei Tang and Huan Liu. Community Evolution in Dynamic Multi-Mode Networks. *Proceedings of Knowledge Discovery and Data Mining*, pages 677–685, 2008. 153

[129] Louis Thurstone. The Vectors of Mind. *Psychological Review*, 41:1–32, 1934. 23

[130] Michael Tipping and Christopher Bishop. Mixtures of Probabilistic Principal Component Analysers. *Neural Computation*, 11:443–482, 1999. 29, 32

[131] Michael Tipping and Christopher Bishop. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society, Series B*, 21:611–622, 1999. 28

[132] Matthew Turk and Alex Petland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 13(1):71–96, 1991. 28

[133] Dimitris G. Tzikas, Astridis C. Likas, and Nikolaos P. Galatsanos. The Variational Approximation for Bayesian Inference. *IEEE Signal Processing Magazine*, 25:131–146, 2008. 43, 44

[134] Naonori Ueda, Ryohei Nakano, Zoubin Ghahramani, and Graham Hinton. SMEM Algorithm for Mixture Models. *Neural Computation*, 12:2109–2128, 2000. 59, 60

[135] Jaakko Viinikanoja, Arto Klami, and Samuel Kaski. Variational Bayesian Mixture of Robust CCA Models. *Machine Learning and Knowledge Discovery in Databases European Conference*, pages 370–385, 2010. 32, 140

[136] Ulrike von Luxburg. A Tutorial on Spectral Clustering. *Statistical Computing*, 17:395–416, 2007. 137, 159

[137] Chong Wang. Variational Bayesian Approach to Canonical Correlation Analysis. *IEEE Transactions on Neural Networks*, 18 Issue 3:906–910, 2007. 31, 32, 139

[138] Robert Ware and Frank Lad. Approximating the Distibrution for sums of Products of Normal Variables. *Technical Report: University of Queensland*, 2003. 142, 143

[139] Duncan Watts. *Small Worlds The Dynamics of Networks between Order and Randomness*. Princeton, 1999. 91, 93, 94

[140] Duncan Watts and Steven Strogatz. Collective Dynamics of 'Small-World' Networks. *Nature*, 393:440–442, 1998. 93, 97

[141] Benhuai Xie, Wei Pan, and Xiaotong Shen. Penalized Mixtures of Factor Analyzers with Application to Clustering High-Dimensional Microarry Data. *Bioinformatics*, 26:501–508, 2010. 48