Probabilistic Models in the Biomedical Sciences

Luke Andrew Carrivick

A dissertation submitted to the University of Bristol
in accordance with the requirements of the degree of
Doctor of Philosophy in the Faculty of Engineering
Department of Engineering Mathematics

October 2005

Abstract

Probabilistic, and in particular Bayesian, methods for modelling data are becoming increasingly sophisticated. This has been fuelled by the demand to analyse the enormous wealth of data being produced by the biomedical sciences. In this thesis we present a variety of unsupervised generative probabilistic models loosely based around mixtures of distributions. The motivation behind using these models is that the mixture reflects aspects of a biomedical process which has a number of contributing factors. We analyse gene expression data from microarray, sequence motif data and radiological data. We attempt to model the interactions between motif data and gene expression for yeast, and we perform in depth analysis of gene expression data for four breast cancer datasets. The radiological data comes from computed tomography scans and radiologist reports. We model the interaction between image data from scans and textual data from reports for a number of lung diseases. A common theme throughout this thesis is *data fusion*: this can be the joint modelling of two separate datasets, comparison of equivalent data sets from independent sources or simply the incorporation of external information into the model.

Acknowledgements

Firstly, I would like to acknowledge the hard work of Colin Campbell who has enthusiastically provided the majority of my supervision for this thesis.

In addition I would like to thank my numerous and helpful collaborators: John Malone in Bristol, Mark Girolami and Simon Rogers at the BRC Glasgow, Colin Cooper and Jeremy Clarke at the Institute of Cancer Research, and Sanjay Prabhu and Paul Goddard at the Bristol Royal Infirmary. Also I am grateful to Nigel Collier at the NII, Tokyo for kindly hosting me during the summer of 2003 and to Jonathan Rossiter for allowing me to commence this research in the first place.

Finally, on a non-academic note I wish to thank all my friends and family for making the task much more enjoyable.

The work presented in this thesis has been completed by the author under sponsorship from UBH Charitable Trust/EMAT.SM6065.6525.

Author's Declaration

I declare that the work in this dissertation was carried out in accordance with the regulations of the University of Bristol. The work is original except where indicated by special reference in the text and no part of the dissertation has been submitted for any other degree.

Any views expressed in the dissertation are those of the author and in no way represent those of the University of Bristol.

The dissertation has not been presented to any other University for examination either in the United Kingdom or overseas.

Signed: .....

Date: .....

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Data Types

In this section we shall introduce the types of data we will be analysing later in this thesis. All the data is taken from the biomedical sciences, in particular from the fields of bio-informatics and medical informatics. We do not attempt to provide an in depth background to the fields of molecular biology and radiology but merely to provide enough information to give a justified motivation for the work in this thesis.

1.1.1 Microarray and Motif Data

In every living organism there are four major types of molecule: deoxyribonucleic acid (DNA), proteins, small molecules (eg. amino acids) and RNA. It is in the DNA and RNA where the genetic *code* for that individual is held. Each time a cell divides this DNA is replicated and passed on to the daughter cell. DNA is constructed as a single or double stranded polymer of the nucleotides: Adenine (A), Guanine (G), Thymine (T) and Cytosine (C). In the case of double stranded DNA, specific pairs of nucleotides form weak bonds together. In terms of bond pairs, $A \leftrightarrow T$ and $C \leftrightarrow G$. This weak bonding means that two complementary strands of DNA will form a stable structure known as the DNA double helix. A consequence of this complimentary bonding is that either strand of a double stranded DNA can be reconstructed

from the other. Much of the structure and internal workings of living organisms is built around proteins. Each protein is a polymer made of a sequence of amino acids. There are 20 amino acids, made up of a triplet of RNA codons labelled U, C, A, G. With $4^3 = 64$ possible combinations there is a many codon triplet to one amino acid mapping. Like DNA, RNA is also a polymer of nucleotides but with Thymine being replaced by Uracil (U), this slight variation in bases means that RNA is always single stranded. RNA does however bind to a complimentary strand of DNA with $A \leftrightarrow U$. This is a very important property which is used in biotechnology.

An individual gene is a stretch of DNA which, under appropriate conditions, manufactures one or more proteins. The Central Dogma of molecular biology, sometimes called Crick's Central Dogma (named after Francis Crick who first coined the term in the 1950's), is a general assumption about the flow of genetic information. It states that sequential (genetic) information cannot be transferred from a protein to either a protein or a nucleic acid. Closely related to this is the overall model for directional information flow in molecular biology. This can be summarised as

$$DNA \rightarrow RNA \rightarrow Protein$$

A fuller model (image taken from [2]) is given in figure 1.1.

This is broken down into four distinct stages: **Transcription**, **Splicing**, **Translation** and **Replication**.

- **Transcription**: This is the process by which a section of DNA is used as a template in the production of messenger RNA (mRNA). Hence information is transferred as $DNA \rightarrow RNA$. This is mediated by RNA polymerase and the relevant transcription factors.
- **Splicing**: This is an additional stage of processing. After **transcription** we have a stretch of what is called pre-mRNA (this unprocessed or partially-processed messenger RNA is called "pre-mRNA" or "hnRNA"). This is then modified to remove certain stretches of non-coding sequences called **introns**. The remaining code includes protein-

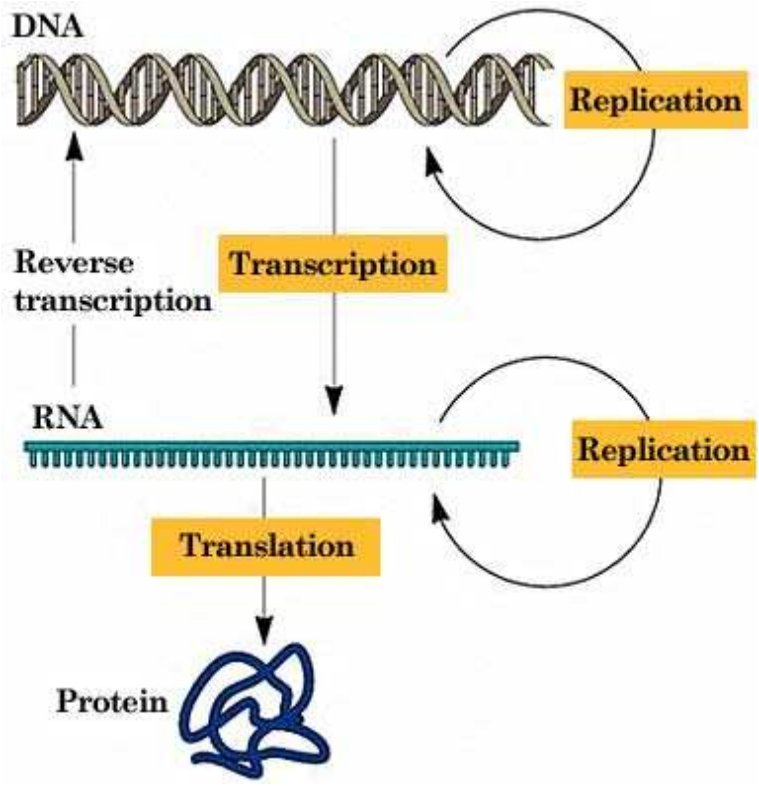


Fig. 1.1: Information flow in Molecular Biology.

coding sequences and are called **exons**. Sometimes one pre-mRNA message may be spliced in several different ways, allowing a single gene to encode multiple proteins. This process is called alternative splicing.

- **Translation**: With the processed mRNA in hand the final stage is to convert this information into the amino acids that make up a protein. In translation, mRNA along with transfer RNA (tRNA), and ribosomes, work together to produce proteins.
- **Replication** Finally, this is the mechanism by which copies of the master template are made. Proteins unwind the double stranded helix and then, through the action of DNA polymerase, we are left with a copy of the original

The aim of Microarray technology is to try and measure the level of activity (ultimately the level of protein production) of individual genes. The motivation for such measurements is to indicate the differences in gene activity under different conditions, eg. Yeast gene activity under stress testing or gene activity in normal and diseased cells. Although the level of mRNA will not necessarily exactly match that of its target protein (the relationship if far from trivial due to variable protein production rates and half lives of mRNA) it is accepted that this will provide a good indication of protein production. It is the mRNA that is measured in a *microarray* experiment.

There are two main techniques for measuring the level of mRNA in a sample, these are *Spotted Microarrays* and *Oligonucleotide Microarrays*. Analysis of cells with *Spotted Microarrays* will give a level of mRNA relative to control whereas *Oligonucleotide Microarrays* require no control. The procedures are summarised below:

- **Spotted Microarrays** : On each spotted array many copies of the DNA sequences of genes are printed at specific locations (these are the spots). Samples of mRNA are then taken from the target cell and a separate control cell, using these two complimentary DNA (cDNA) sequences are constructed. To distinguish the target cDNA and control cDNA they are each treated with different fluorescent dyes. When this resulting cDNA is combined and hybridised over the microarray sections of it will bind to the corresponding DNA at specific locations on the array. The array is then washed to remove non-specific binding. A laser is then used to determine the intensity of each dye at each

spot. This is a measure the level of hybridisation for both the target and control sample for each gene. It is common to use the log of the ratio of the two levels to maintain a symmetry between over and under expression of genes. An example Spotted Array is given in figure 1.2. In this case red corresponds to over expression compared to the control cell and green corresponds to under expression. Additionally black indicates poor hybridization and yellow an even level of hybridization between the target and control cell.

- **Oligonucleotide Microarrays** : For these arrays the underlying principle and motivations are the same but the practical implementation is somewhat different. These Microarrays consist of Oligonucleotide (these are 25 nucleotide long polymers) that are positioned on a silicon array at specific sites called probes. The Oligonucleotides are chosen to match DNA subsequences of the genes which we are measuring the activity for. Each microarray will have up to a million unique Oligonucleotide probes. Each gene corresponds to at least one set of 11 different probe pairs. Probe pairs are made up of a of a 25-base-pair perfect-match (PM) Oligonucleotide probe and a 25-base-pair mismatch (MM) probe. The mismatch probe is identical to the match probe but has the central nucleotide reversed. Once again a cDNA sample is washed over the microarray and the level of hybridisation at each probe measured. The information from each probe can then be combined to give an expression value for each gene. By analysing the difference in signal of the PM and MM probes a statistical confidence, known commonly as the p-value, in the overall expression can be given. Additionally a rating of absent (A), present (P), or marginal call (M) is given for each gene based on significant deviation in expression from zero. Oligonucleotide Microarrays are a newer technology and have shown to be more accurate than Spotted Arrays. For more information see [3].

A *Sequence Motif* is a string of amino-acids or nucleotides which has been deemed biologically significant. Significance can be determined by experimental or statistical means. Motifs found within the exon of a gene are thought to encode common structural elements of the final protein (structural motifs). *Regulator Sequence Motifs* are found outside of the exon, these are sequences where transcription factors preferentially bind and it is thought that, rather than encoding common structural elements, they influence the shape of the final protein. In chapter 5 we shall look at the correspondence between gene expression and Regulator

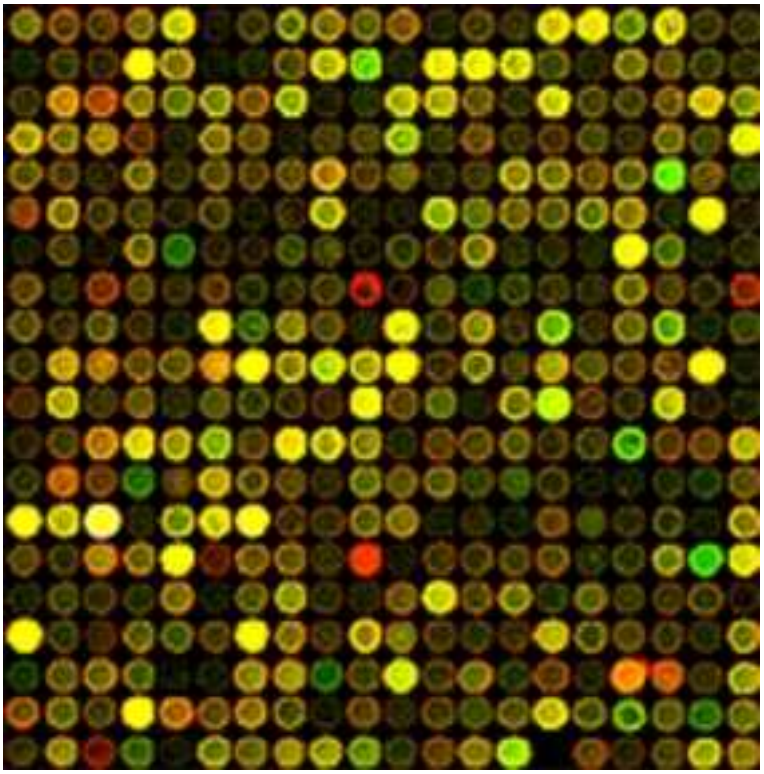


Fig. 1.2: Dyes indicate the level of complimentary binding to a particular sequence.

Sequence Motif abundance. Additionally there are *Short coding motifs* which lack any relationship to structure in the final protein. Motifs need not be exact strings, they can contain wild cards and logical operators. We shall give a well know example, the *N-glycosylation* Motif, using the standard amino acid abbreviations:

Asn, anything but Pro, Ser or Thr, anything but Pro

1.1.2 Radiological Data

Computed Tomography (CT) or Computed Axial Tomography (CAT) is a method of medical imaging that generates a 3D internal picture of the body. As a technique, CT was invented in the early 1970's by Godfrey Hounsfield who subsequently went on to share the Nobel Prize in 1979 for his work in this area. The first CT scanners were used to image sections of the brain, since then the technique has been extended to image all other areas of the body. There have been many generations of improvement in the technology with improved speed, detail and a lowering of the radiation dosage needed. However, CT scans still expose the patient to radiation several times that of an x-ray so will often be reserved for more seriously ill patients. CT scanners work by rotating, in a circular fashion, an x-ray around the area of the body which is to be imaged. Sensors on the opposite side of the circle then detect the intensity of the received x-rays. This data is the collated and then using tomographic reconstruction a series of 512×512 images are generated. Each 512×512 slice is a matrix of values representing the density of tissue at a single point. The pixel density is measured in the Hounsfield units which are calibrated from -1000 for air to +1000 for bone. Three examples of CT images of the chest are given in figures 1.3, 1.4 and 1.5. It is now commonplace to use CT as method of imaging for many areas of the body. In this thesis we will restrict ourselves to CT images of the chest. CT is particularly well suited to the detection and diagnosis of both subtle and chronic changes in the lung parenchyma. In a number of cases, such as cancer and pneumonia, it is necessary to give a contrast agent to the patient to increase or decrease the intensity of certain aspects of the resulting scan. The thickness and detail of the scans can be changed depending on the nature of the suspected disease. For example, chronic interstitial processes such as emphysema and fibrosis require thin sections with high spatial frequency reconstructions. Good introductions to Chest Radiology and the respiratory system are given

in [46, 65].We will now give a brief overview of lung diseases that are diagnosed via CT and a number of examples of chest scans with common disease types.

CT generates more detailed images than conventional x-rays and so is often used to first detect tumours in the lung region or mediastinum (the area between the lungs). It also can be used as a repeat technique to monitor a cancerous tissues response to treatment; this is known as nodule tracking. Pneumonia and tuberculosis are also visible on a CT scan. Figure 1.3 shows CT scans taken from 3 patients all suffering from fibrosis. Fibrosis is a very general term and either refers to scar tissue that forms as a consequence of another disorder, or, where there is no known cause, it is called *Idiopathic Pulmonary Fibrosis*. Fibrotic tissue is denser than normal tissue and so will appear whiter on a CT scan. It is often described as *reticular*, that is forming a mesh like structure, and in the latter stages of fibrosis the lung can take on a honeycomb appearance. Another common lung disorder is Emphysema. This is a permanent formation of air sacks in the lung field, these air spaces will appear as dark patches on the CT scan as they have a very low density. Consolidation in a lung is an area that contains fluid or other material which should normally contain air. This will generally appear as denser, that is whiter on the CT image. A new classification of lung disorder that is not visible on x-rays is Ground Glass Opacification. This is defined by [32] as "hazy areas of increased attenuation without obscuration of the underlying vessels". It is indicative of early stage consolidation or fibrosis.

The first example in figure 1.3 shows the early stages of fibrosis, which is characterised by a faint *wispy* appearance, particularly toward the perimeter of the lung field. In the second example this patient is suffering from severe fibrosis in combination with bullous Emphysema. The third example shows the honeycombing of the lung that is sometimes associated with the latter stages of fibrosis. The first two example in figure 1.4 both show bi-lateral bullous Emphysema in the posterior (toward the back of the patient, so if the CT is taken with the patient lying on their back this will appear at the bottom of the image) of the lung. The first example also show over expansion of the lung field on the right which is often associated with Emphysema. In the third example the Emphysema is mainly located in the anterior of the left lung. Figure 1.5 gives three examples of Ground Glass Opacification, in the first of these images the opacification is present uniformly across the lung field but in the remaining examples the opacification is more localised.

Fig. 1.3: Three examples of Fibrosis. The first image shows a patient in the early stages of the disease, while it is more fully progressed in the remaining images.

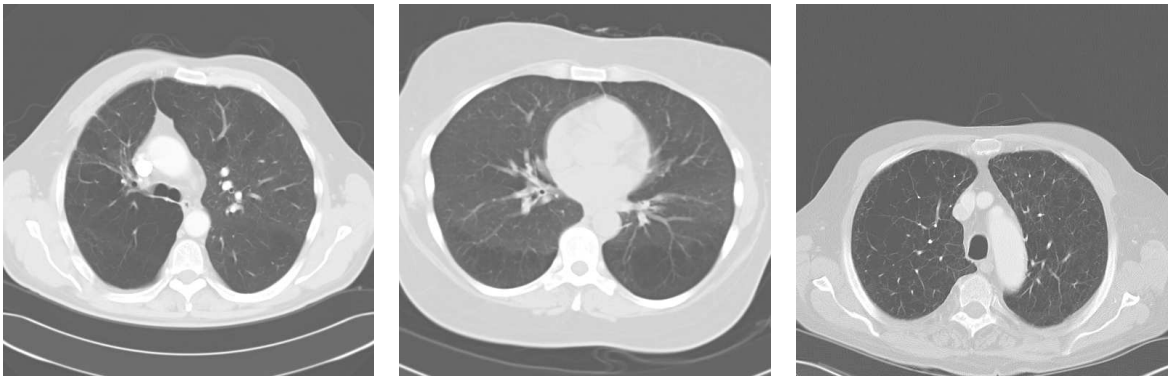


Fig. 1.4: Three examples of Emphysema. This is distinguished by darker patches in the lung field.

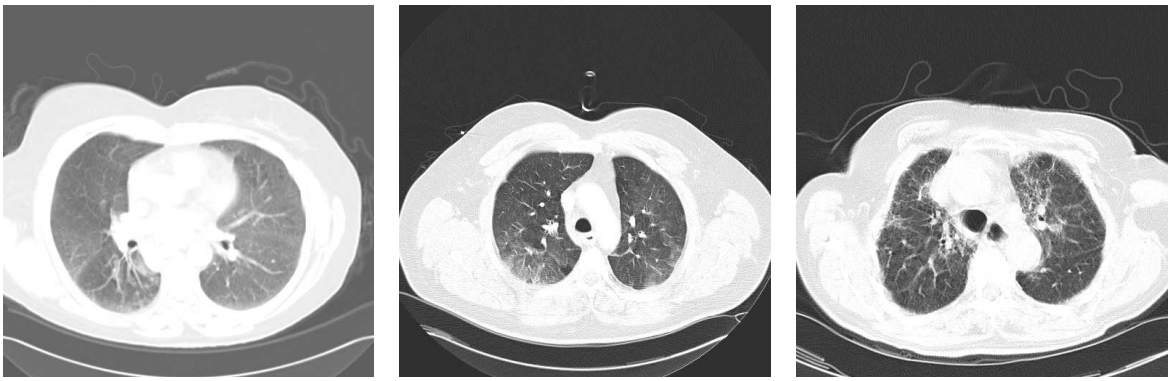


Fig. 1.5: Three examples of Ground Glass Opacification

1.2 Machine Learning

In this section we shall give an overview of some popular and relevant methods of Machine Learning and indicate where they have been previously applied to problems in Medical and Bio-Informatics. We shall put particular emphasis on probabilistic techniques as they will form the basis of this thesis.

Machine learning algorithms are techniques by which computers can mechanically learn to do a task. It is often hoped that during this process new insights into the data or the broader field from which the data is taken may be gained. The algorithms fall broadly into two classes: Supervised and Unsupervised.

In supervised algorithms there is a substantial human input to the problem, for example this could be labelling of data points. Classification algorithms are the most well know supervised techniques.

In most practical applications of classification the data is partitioned into a training and testing set. The test set is used to verify the accuracy of the learnt model and to avoid over-fitting. In unsupervised algorithms the emphasis is shifted toward discovery of patterns within the data. Clustering is the most popular unsupervised technique. Additionally there is a semi-supervised class of algorithms concerned with reinforcement learning, none of which we shall not mention here. Learning algorithms can also be classed as being parametric or non-parametric. Roughly speaking non-parametric methods make no assumption about the statistical distributions within the data.

The most simple and well known probabilistic classifier is the Naive Bayes algorithm [54]. In this method all features, that is each element of a single data point, are assumed to be independent given a class label. Class dependent parameters are estimated using the training data and classification of the test data is done via Bayes Rule. Despite its seemingly over simplistic assumptions Naive Bayes is popular and has had success in many applications.

One of the most successful classifiers is the Support Vector Machine (SVM) [84]. This is a non-parametric technique most suited to binary classification. It works by constructing an

optimal, in the sense of maximally separating, hyperplane between the classes of points, this hyperplane is defined by using only closest data points which are known as the *support vectors*. It is in effect concentrating on modelling the boundary between classes rather than the classes themselves. Classification of new unseen data is easy and done by computing which side of the hyperplane the test point lies. It is attractive as an approach since with separable datasets a unique optimal hyperplane exists, which is determined by convex quadratic programming problem. SVM's are generally regarded as the successor to neural networks [14]. They are an example of a Kernel Method, thus utilising the so called *Kernel trick*. In its simplest form with a linear hyperplane the quadratic programming problem is dependent only the matrix of dot products between each data point. Through Mercer's condition, which states that *any positive semi-definite kernel K(x, y) can be expressed as a dot product in a high-dimensional space*, simple dot products can be exchanged for more complex non-linear kernel functions. This non-linear mapping is never explicitly calculated as we are only every interested in distances in the new space. In the case of SVM's this non-linearity can be thought of as either a non-linear (curved in feature space) separating hyperplane, or alternatively a non-linear mapping of feature space to a higher dimension in which the hyperplanes remains flat. The form of this mapping is determined by the Kernel function, which can take on many forms depending on both the complexity of the problem and the nature of the data, eg. RBF Kernels for numerical data or String Kernels for textual data.

A closely related classifier to the SVM but constructed from a more Bayesian perspective is the Bayes Point Machine (BPM) [42]. The motivation behind SVMs is easily understood by considering hyperplanes in the vector space defined by the dimensions of the data. By working in the dual space of this, known as version space, BPMs can be more easily visualised. In this dual space hypotheses (that is separating hyperplanes) become points and the data points become hyperplanes. The SVM solution is to pick the point (hypothesis) at the centre of the largest sphere which can fit into the space defined by the hyperplanes. Hyperplanes tangent to this sphere are the support vectors. The Bayes Point solution finds the midpoint of the region of intersection of all the hyperplanes. This is in effect using all the data available, not just the support vectors. As a method it has been shown to give promising results but as it is extremely hard to compute the Bayes Point it does not have widespread usage.

The Relevance Vector Machine [80] is another Bayesian approach to classification. Like the SVM it is a sparse kernel method. It has the advantage of providing a full predictive

distribution, rather than point values (as in the SVM case). The *Relevance Vectors* are constructed as the most informative examples in the training data, like the Bayes Point Machine it has not received the widespread popularity that the SVM has.

There are two major classes of clustering technique: hierarchical clustering and partitional clustering. The most common hierarchical techniques are agglomerative methods. These algorithms work by firstly initialising each data point into its own cluster, and then sequentially merging the most similar points until there is only a single cluster containing all the data. This gives rise to a tree like structure which is often referred to as a dendrogram.

It is most common in partitional clustering to form a predetermined number of groupings in the data. K-Means ([29]) is perhaps the most common partitional clustering technique. It is an iterative technique which works by first randomly (or by some other means) picking the cluster centres, each point in the data set it then assigned to the nearest cluster centre. The centres are then recomputed and the algorithm repeats until it has converged to a stable solution. The results can be greatly influenced by the initial assigned starting positions of the cluster centres. An extension to K-means is fuzzy K-means [12] this is simply the case where each point has a graded membership to each cluster rather than a binary one which is the case with standard k-means. Both of these methods can be though of as special cases of Mixture Models ([14]) which are generally regarded as the most mature of all clustering approaches. Mixture models are a class of probabilistic clustering methods that attempt to model datasets as combinations of distributions. A very recent clustering technique, which can be though of as a generalisation of standard Mixture Models is Latent Dirichlet Allocation [15] (LDA). We shall use the LDA framework extensively in this thesis. A more thorough discussion of Mixture Models can be found in section 2.2.

1.2.1 Machine Learning for Microarrays

There are two distinct areas where machine learning, and in particular statistical techniques, have been used to analyse microarray data. The first is a statistical analysis of the reliability of the data itself. This is very important as the technology is new and produces inherently noisy results. The second is tertiary analysis of the datasets assuming the results are accurate, this

is an area where both unsupervised and supervised machine learning have been successfully applied. In this thesis we shall concentrate on the latter.

The development of Microarrays gave a new insight into the activity of genes. For example, this allowed a more detailed differentiation between cells which would otherwise have looked identical. For this reason unsupervised techniques are the most important in the analysis of Microarrays. Perhaps the first paper to apply unsupervised techniques to Microarrays was Eisen et al [30]. This used hierarchical agglomerative clustering to discover functional gene groupings in Microarrays taken from yeast. These groupings of were genes were found to be associated with cell cycles. The accompanying software has been widely used in analysis of other data sets.

One application of microarray analysis has been in the discovery of genetically distinct subtypes of disease [5, 25, 40, 41, 68, 77, 83, 87]. This has often been done with reference to an external measure, for example a correspondence between disease subtype and patient survival times in cancer. These discoveries point toward more targeted or preferential treatment of patients based on sub type association.

Another application of supervised machine learning has been in cellular classification. This is concerned with classification of cells, eg. samples from different patients, into a number of pre-determined classes. A common example would be classification of healthy vs diseased, though a clear diagnostic aid the main worth of this work likes in analysing the classifier to determine which genes or groups of genes distinguish healthy from diseased. For example Li *et al* [55] use a RVM classifier to classify normal against diseased samples on a number of cancer data sets.

Another application has been of classification of the genes themselves. For example in [17] the authors used an SVM to classify six functional classes of yeast gene based on expression values over a number of experiments. This work highlighted genes which are consistently classified well, indicating a well defined functional class and genes which are consistently classified poorly indicating a need to investigate their functionality further.

In chapters 5 and 6 we shall investigate microarray expression data using a number of machine learning techniques.

1.2.2 Machine Learning in Radiology

From the earliest rule based decision support systems, [75, 43], machine learning has been applied to problems in the medical domain. Radiology has seen two main areas where machine learning techniques have been used, the first is the classification and processing of consultants reports [35, 79]. This work is of general interest to the Natural Language Processing (NLP) community. One of the main motivations for automatically processing reports is in retrieval from medical databases. The second area of interest is in the processing of data from the CT or X-Ray images themselves. Automatic segmentation, that is separating an area of the image that has a predefined function, is often seen as the first step in a medical image processing problem. Numerous methods have been proposed for automatic pulmonary image segmentation, these include model based approaches, pixel thresholding, region growing and pattern recognition based (see [44, 33, 64] for some examples). In [64] a Gaussian mixture model is used to partially segment MRI images of the brain. In this thesis perfect segmentation is not relevant, and so we do not concentrate on this aspect of the image processing and use a simple thresholding method to extract the lung fields. Texture classification of lung CT images is a popular area of research ([76, 81]. The motivation for this is to provide a accurate, reliable and, very importantly, a reproducible way of automatically labelling a variety of diseases on a lung CT scan. In Uppaluri *et al* [81], the authors use a Naive Bayes classifier to detect *honeycombing, ground glass, bronchovascular, nodular, emphysema* and *normal* tissue types.

In chapters 3 and 4 we shall perform unsupervised learning on Radiological Data.

1.2.3 Data Fusion

In the chapters 4 and 5 of this thesis we simultaneously model two different types of data, in chapter 3 we incorporate *expert knowledge* into our model and in chapter 6 we contrast the results of modelling different, but directly compatible, datasets. These approaches can all be viewed, on some level, as forms of *Data Fusion*. We shall now give a brief overview of *data fusion* relevant to this thesis.

In the paper by Segal *et al* [72] the authors fuse data from a number of DNA microarray experiments into one single data set. This contains 1,975 samples taken from cancer tissues covering 22 different tumour types. As the original experiments which generated the data are independent there are many factors which make direct comparison of expression values hard, and as there is variation in the actual genes whose expression is measured between studies there are many missing values. None the less some tentative but coherent processes (which they refer to as modules) that are common to more than one cancer type were discovered in this study. In Sorlie *et al* [78] they study a number of Breast Cancer DNA microarray data sets and find a number of common subtypes between each one. See chapter 6 for related work. In the paper of Middendorf *et al* [61] they try to simultaneously model gene expression values, expression levels of regulators and up stream transcription factor motif abundance. This allows prediction of expression levels based on expression levels of regulators and motif profiles. Related work exists in [73], where the authors propose a probabilistic model for initial selection of the motifs. In chapter 5 we shall study the same yeast dataset originally published in Gasch *et al* [36]. This has been further analysed in both [61] and [73]. Another important and recent advancement in the field of *Data Fusion* has been in the field of Kernel Methods. Kernel methods are, on the whole used, for classification and regression but very occasionally are used for clustering ([11]). Kernel functions $K(x, y)$ ([74]), are be constructed to handle different types of data, for example, diffusion Kernels for graphical data [48] and string kernels for sequence data. One way to perform data fusion is to construct a composite Kernel that is a linear combination of Kernels $\mathcal{K}(x, y) = \sum_i \beta_i K_i(x, y)$. Thus disparate types of data can be combined into a single Kernel. A variety of techniques have been suggested to learn the weighting parameters β_i, for example Lanckriet *et al* ([53]) use a semi-definite programming approach while Girolami *et al* ([37]) use a hierarchical Bayesian model.

1.3 Publications

1. *Unsupervised Learning in Radiology Using Novel Latent Variable Models*. Luke Carrivick, Sanjay Prabhu, Paul Goddard and Jonathan Rossiter.**IEEE Computer Vision and Pattern Recognition** (CVPR), San Diego, California USA, June 2005. Pages 854-859.

2. *Deriving a Hierarchical Representation of Lung Disease using Re-Sampling Mixture Models*. Luke Carrivick and Sanjay Prabhu. **Medical Image Understanding and Analysis** (MIUA). Bristol, UK, July 2005. Pages 155-158.

3. *Identification of Prognostic Signatures in Breast Cancer Microarray Data using Bayesian Techniques*. Luke Carrivick, Simon Rogers, Jeremy Clarke, Mark Girolami, Colin Campbell and Colin Cooper. **Journal of the Royal Society** *Interface*. (In press).

4. *Prognostic Expression Signatures for Human Breast Cancer* Luke Carrivick, Simon Rogers, Jeremy Clark, Mark Girolami, Colin Campbell and Colin S Cooper. **Genes Chromosomes and Cancer**. 2005.

1.4 Glossary

In this thesis we shall often refer to the term process. A process is defined by a set of parameters for specified distributions and is equivalent to the more commonly used term *Mixture*. We do no use the term mixture to distinguish the models used in this thesis from Mixture Models. Throughout this term *sample* is used in two different contexts. Firstly as indicating an element of a data set eg Sample meaning an individual patient from a cohort, and secondly as indicating the result of sampling from a probability distribution. Here we shall give a glossary of the variable conventions used in chapters 3, 4, 5 and 6.

Convention for indices:

- k - a process index and has limits $1 \ldots K$.
- d - in chapters 3 and 4 an index for CT image / Report parings.
- n - a region index for an individual CT image.
- f - a feature index for an individual CT region.
- m - in chapter 4 a word index for radiology reports.
- m - in chapter 5 a motif index under the multinomial model.
- n - in chapter 5 a motif index under the Poissonian model.
- g - a gene index.
- d - an experiment index in chapter 5.
- d - an sample (patient) index in chapter 6.

Convention for data:

- R_{ndf} - the numerical value of feature f, in region n of CT image d.
- W_{md} - the count of word w in radiology report d.

- M_{mg} - the index of the m'th motif in gene g.
- C_{ng} - the overall count for motif n in gene g.
- E_{dg} - the expression of the the g'th gene of patient d.

Convention for Model Variables:

- $\boldsymbol{\Theta}$ - the set of all model parameters.
- $\boldsymbol{\alpha}$ - a k dimensional Dirichlet parameter.
- μ - the mean of a Gaussian distribution.
- σ^2 - the variance of Gaussian distribution.
- β - in chapter 4 a multinomial of words.
- β - in chapter 5 a Poisson parameter for motif counts.
- ν - in chapter 5 a multinomial over motifs.

A standard function used in this thesis is the digamma function Ψ, defined as:

$$\Psi(x) = \frac{\partial \log\left(\Gamma(x)\right)}{\partial x}$$

Additionally there are many latent variables used in this thesis, these will be defined wherever used.

Chapter 2

Probabilistic Models

2.1 Probabilistic Modelling

In this chapter we present some of the background material and methods behind probabilistic modelling. Some of which we shall call upon in later chapters.

2.1.1 Approaches: Bayesian and Frequentist

There has been a huge amount of literature and debate surrounding the differences between *Bayesians* and *Frequentists*. Here we shall not address any of the philosophical differences between the two fields of thought as this is often irrelevant to the application of the methods. But instead we shall just present the motivation behind each approach, and demonstrate this using observations from a simple discrete random variable. It is often the case that the only distinction between the two approaches is made by the method in which results are presented and the conclusions which can then be drawn.

Suppose we have a number of iid observations $X_1, \ldots, X_N$ each of which can take a value $1, \ldots, k$.

In a *Frequentist* approach the belief is that given enough data, in the limit $N \to \infty$, we can

calculate the true value of

$$P(X = i) = \frac{\sum_n \delta(X_n, i)}{N}$$

Where $\delta(X_n, i) = 1$ if $X_n = i$, and so summing this gives an overall count of the number of observations for a particular i. The goal for a *Frequentist* is to calculate a interval, whose centre is on the sample mean, in which they are confident the true value of $P(X = i)$ lies. Typically you would want to give a small interval and a high confidence (at least 90%) for the estimated parameters.

In a *Bayesian* approach the emphasis is a step further back. Not only are the observations taken as random variables from a distribution but the parameters governing these distributions are also, themselves, random variables. The goal for the *Bayesian* is therefore to estimate the distribution of $P(X = i)$. Thus as the *Bayesian* has a full distribution for the parameters, a stronger conclusion can be drawn in that they can say *There is a 90% probability that the* $P(X = i)$ *lies in an interval.* This approach of treating parameters as random variables can be extended to the hyper-parameters of the distributions governing the parameters, and so on. The *Bayesian* approach allows for incorporation of prior knowledge in a very explicit way. If A is our set of knowledge about the world, this could be parameters of a distribution, and B are some new observations then we can, in light of the new data, update our distribtion $P(A)$ using *Bayes* rule in the following way:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.1}$$

The left hand side is a distribution so must integrate (or sum in the case of discrete distributions) to one, because of this it is often more convenient to consider the simpler form:

$$\begin{aligned} P(A|B) &= \frac{P(B|A)P(A)}{\sum_{A'} P(B|A')P(A')} \\ &\propto P(B|A)P(A) \end{aligned} \tag{2.2}$$

Here $P(A)$ is the current or *prior* distribution for A, $P(B|A)$ is the *likelihood* and $P(A|B)$ is the *posterior*. Bayes rule, in equation 2.2, can be iteratively updated each time new data is received.

We shall now present a well known, straight forward, but demonstrative example of using Bayes rule. The aim is to make a general point about false positives and negatives in any kind of test.

Suppose we have a medical test for a particular type of cancer. This test is, or at least appears to be, on the whole very accurate. It gives:

- For a patient who actually has the cancer the test will detect it, a positive result, with a probability 0.99. That is 99% of the time.
- For a patient who is healthy the test will show a clear, negative result with probability 0.95.

In addition the particular cancer is rare, occurring in only 0.5% of the population. If we denote P as a positive result, N as a negative result, D as actually having the disease and $\hat{D}$ as not having the disease then we have now got the following:

$$\begin{aligned} P(P|D) &= 0.99 \\ P(N|\hat{D}) &= 0.95 \\ P(N|D) &= 1 - 0.99 = 0.01 \\ P(P|\hat{D}) &= 1 - 0.95 = 0.05 \\ P(D) &= 0.005 \\ P(\hat{D}) &= 1 - 0.005 = 0.995 \end{aligned} \tag{2.3}$$

Using *Bayes'* rule from equation 2.2 we can now calculate the probabilities of getting false positives or false negatives

$$P(\hat{D}|P) = \frac{0.05 \times 0.995}{0.05 \times 0.995 + 0.99 \times 0.005} = 0.9095 \tag{2.4}$$

$$P(D|N) = \frac{0.01 \times 0.005}{0.01 \times 0.005 + 0.95 \times 0.995} = 5.2893e - 05 \tag{2.5}$$

The result for a false positive $P(\hat{D}|P)$ can appear to be somewhat counterintuitive. It says even with a positive test result the chances are you are healthy. This is because the probability of a test failing is higher that the probability of actually having the cancer. It shows that for tests on rare diseases to be useful they must be very accurate.

2.1.2 Exchangeability

The concept of exchangeability was first formalised in a theorem by De Finetti's in the 1930's. It stated that, for an infinite series of random variables

$$X_1, X_2, X_3, \ldots$$

to be exchangeable then for a finite sequence $X_{i1}, \ldots, X_{in}$ permuting the indices to give a reordered sequence will leave the probability distribution unchanged.

2.1.3 Conjugate Priors and The Exponential Family

In a Bayesian approach to probabilistic modelling the *posterior* distribution is obtained by multiplying the *likelihood* by a *prior*. Substituting $B = \Theta$ into equation 2.2 and taking A as observed data, the *posterior* of the parameters Θ given the data is.

$$P(\Theta|A) \propto P(A|\Theta)P(\Theta) \tag{2.6}$$

The prior distribution for the parameters $P(\Theta)$, belonging to a family P, is said to be conjugate to the *likelihood* $P(A|\Theta)$ if the *posterior* $P(\Theta|A)$ remains in the family P. One property

of this is that it allows easy sampling of the *posterior* from well understood distributions. We shall see later (section 2.3.7) that this is also particularly useful when performing sampling based inference.

An important concept in probabilistic modelling is the concept of sufficiency in a statistic. If we have number of observations $\mathbf{x} = \{x_1, \dots x_n\}$ taken from statistical model F with parameters θ a function of those observations, $T(\mathbf{x})$, is said to be sufficient if

$$P(x|T(\mathbf{x}) = t, \theta) = P(\mathbf{x}|T(\mathbf{x}) = t) \tag{2.7}$$

That is to say, all we can know about the unknown parameters θ is captured in the statistic $T(\mathbf{x})$ and having the original $\mathbf{x}$ is of no gain.

The Exponential Family of distributions are the set of probability density functions that can be written:

$$p(x|\theta) = h(x) \exp(\eta(\theta)^\top T(x) - A(\theta)) \tag{2.8}$$

Where $T(x)$ is the vector (or possibly a scalar) of sufficient statistics, $A(\theta)$ is the log partition function and $\eta(\theta)$ is known as the *canonical* or *natural* parameter. Since

$$\int p(x|\theta)dx = 1$$

, we can write

$$A(\theta) = \log \int h(x) \exp(\eta(\theta)^\top T(x))dx \tag{2.9}$$

As a simple example we will show that the *Poisson* distribution is a member of the exponential family.

$$p(x|\theta) = \frac{\theta^x e^{-\theta}}{x!} = \frac{1}{x!} e^{(x \log(\theta) - \theta)} \tag{2.10}$$

So $T(x) = x$, $A(\theta) = \theta$, $\eta(\theta) = log(\theta)$ and $h(x) = \frac{1}{x!}$.

The identification of an Exponential Family class of distributions is primarily a mathematical construction. Many useful quantities can be derived using $T(\mathbf{x})$, $A(\theta)$ and $\eta(\theta)$. Another property of this family of distributions is that for each member of the family there exists a conjugate prior that is also a member of the family. This is easy to show using a simple example.

If we draw a number of observations from a distribution in the exponential family, using the form given in equation 2.8, the likelihood of these observations is given as:

$$p(\mathbf{x}|\theta) = \left[\prod_i h(x_i)\right] \exp(\eta(\theta)^\top \sum_i T(x_i) - \sum_i A(\theta)) \tag{2.11}$$

This likelihood has sufficient statistic $\sum_i T(x_i)$ and the same natural parameter $\eta(\theta)$. It is then possible to construct a prior distribution

$$p(\theta|\alpha, \beta) \propto \exp(\eta(\theta)^\top \alpha - \beta A(\theta)) \tag{2.12}$$

such that the posterior,

$$\begin{aligned} p(\theta|\mathbf{x}, \alpha, \beta) &\propto p(\mathbf{x}|\theta) p(\theta|\alpha, \beta) \\ &\propto \exp(\eta(\theta)^\top (\alpha + \textstyle\sum_i T(x_i)) - (\beta + N) A(\theta)) \end{aligned} \tag{2.13}$$

where $N = \sum_i$. Thus equation 2.13 is in the same form as the prior 2.12. The parameters α and β have been changed to $\alpha + \sum_i T(x_i)$ and $\beta + N$ respectively, but the natural parameter

$\eta(\theta)$ and the log partition function $A(\theta)$ remain unchanged. Conjugate priors are useful in a mathematical sense, but can sometimes be seen as too restrictive on the form of the prior. An classic example of this is the Binomial distribution.

$$P(x = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

for $k = 0, 1, 2, \ldots, n$ A sensible prior would perhaps take the form of a bimodal distribution. However the conjugate prior, a Beta distribution,

$$f(x) = \frac{1}{\mathrm{B}(\alpha, \beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

is not expressive enough to have a bimodal form. A distribution is self-conjugate if it and its conjugate prior come from the same class of distributions. As a well known example of self-conjugacy, we give the Gaussian distribution as a conjugate prior for the Mean of a Gaussian. For simplicity we take a zero mean prior.

$$\begin{aligned} p(\mu|x, \sigma, \tau) &\propto p(x|\mu, \sigma) p(\mu; \tau) \\ &\propto \exp(\tfrac{-(x-\mu)^2}{2\sigma^2}) \exp(\tfrac{-\mu^2}{2\tau^2}) \end{aligned} \tag{2.14}$$

As the posterior is a density for the parameter μ all terms not involving μ can be discarded. Completing the square in the exponential for μ and making the substitution $s^2 = \sigma^2 + \tau^2$ we now have

$$\begin{aligned} p(\mu|x, \sigma, \tau) &\propto \exp(-\tfrac{s^2}{\sigma^2\tau^2}(\mu^2 - 2x\mu\tfrac{\tau^2}{s^2})) \\ &\sim N(\tfrac{x}{1+\sigma^2/\tau^2}, (\tfrac{1}{\tau^2} + \tfrac{1}{\sigma^2})^{-1}) \end{aligned} \tag{2.15}$$

Which is indeed a Gaussian distribution, the same family as the chosen prior for μ.

Another useful consequence of the exponential family formalism is the ease it which you can calculate the moments of the sufficient statistics. If the parameterisation is such that $\eta(\theta) = \theta$ then it is in *canonical* form. If we assume this form and differentiate equation 2.9 with respect to θ_i (reversing the order of integration and differentiation is valid here) then we have:

$$\frac{\partial A(\theta)}{\partial \theta_i} = \frac{\int h(x) T_i(x) \exp(\theta^\top T(x)) dx}{\int h(x) \exp(\theta^\top T(x)) dx} = \int h(x) T_i(x) \exp(\theta^\top T(x) - A(\theta)) dx = E\left[T_i(X|\theta)\right] \tag{2.16}$$

Where X is a random variable belonging to the exponential family. This property for a Dirichlet distribution is used in both the papers of Blei *et al* [15] and Rogers *et al* [70].

2.1.4 Jensen's Inequality

Jensen's inequality, introduced in [45] is of huge use in information theory and probabilistic modelling. There are many guises in which it appears, and we shall only present the most relevant one in this thesis. It states that under a convex function F the image of the expectation of a random variable X is greater than or equal to the expectation of the image of X.

$$F(E(X)) \geq E(F(X))$$

As an simple example, we shall use a discrete probability distribution with density a_i, such that $\sum_i a_i = 1$ and the convex function log. Jensen's inequality is then written as:

$$\log \sum_i i \times a_i \geq \sum_i \log(i) a_i$$

or alternatively the arithmetic mean is always greater than or equal to the geometric mean.

$$\sum_i i \times a_i \geq \prod_i i^{a_i} \tag{2.17}$$

It is interesting to note that Jensen's inequality is an equality if and only if all the probability mass lies at a single point, i.e. $a_j = 1$ and $a_{\neq j} = 0$. This is in reality rarely the case. Jensen's inequality is a useful technique to break the coupling between variables in a summation or integral. This often turns an intractable calculation into a tractable one.

2.1.5 Kullback Liebler Divergence

The Kullback-Liebler divergence [49] is a measure of the similarity between two distributions. For discrete distributions it is written:

$$KL(p||q) = \sum_x p(x) \log \left[\frac{p(x)}{q(x)}\right] \tag{2.18}$$

with an integral replacing the summation for continuous distributions. This is non-symmetric, so in general $KL(p||q) \neq KL(q||p)$. The KL divergence can be expressed in terms of entropies

$$KL(p||q) = \sum_x p(x) \log p(x) - \sum_x p(x) \log q(x) = -H(p) + H(p,q) \tag{2.19}$$

Where $H(p,q)$ is the cross entropy between p and q and H(p) the entropy of p. It is important to note that $KL(p||q) = 0$ iff $p = q$.

2.1.6 Maximum Likelihood and Maximum a Posteriori

Maximum-Likelihood is a classical approach to parameter estimation. Under the frequentest view parameters are taken as having fixed values. Contrastingly in a Bayesian setting pa-

1=121	2=125	3=123	4=114	5=117	6=127	7=126
8=117	9=124	10=129	11=136	12=128	13=102	14=115
15=113	16=107	17=117	18=124	19=127	20=96	21=107
22=118	23=137	24=115	25=143	26=122	27=128	28=127
29=122	30=127	31=133	32=130	33=130	34=116	35=127
36=116	37=114	38=154	39=111	40=133	41=101	42=118
43=145	44=144	45=128	46=121	47=140	48=135	49=118

Tab. 2.1: National Lottery Data

rameters are treated as random variables as so require prior distributions, this formulation gives rise to the Maximum a Posteriori approach to parameter estimation.

The Maximum-Likelihood solution is taken as the set of parameters, Θ_{MAP}, that maximises the likelihood of the data given the model parameters, $P(Data|\Theta)$ (here Θ represents the model parameters). The *Maximum a Posteriori* solution is taken to be the set of parameters that maximises the posterior, $P(\Theta|Data)$, of the parameters given the data. The MAP solution is related to the ML solution though Bayes Rule.

$$P(\Theta|Data) \propto P(Data|\Theta)P(\Theta) \tag{2.20}$$

Thus the MAP solution enables us to incorporate prior knowledge about the distribution of the parameters values. As MAP solutions are generally derived algebraically it is not as essential to use conjugate priors than would be the case in sampling based inference.

A simple example is taken from the UK National Lottery [4]. Table 2.1 shows the frequency with which the numbers have appeared from the first draw up to August 6th 2005.

We assume the frequency, x_i, with which a number appears forms a multinomial distribution with parameters θ_i. The likelihood of the the data given the parameters is therefore.

$$L(Data|\Theta) = \frac{49!}{\prod_{i=1}^{49} x_i} \prod_i \theta_i^{x_i}$$

The log-likelihood is

$$\log L(Data|\Theta) = \log(49!) - \sum_{i=1}^{49} \log(x_i!) + \sum_{i=1}^{49} x_i \log(\theta_i) \tag{2.21}$$

To maximise the log-likelihood, and hence likelihood, we differentiate equation 2.21 with respect to the model parameter θ_j with the additional constraint that $\sum_i \theta_i = 1$.

$$\frac{\partial}{\partial \theta_j} \left(\log L(Data|\Theta) + \lambda(\sum_i \theta_i - 1) \right) = \frac{x_j}{\theta_j} + \lambda) \tag{2.22}$$

Where λ is a Lagrange multiplier. Thus:

$$\theta_j^{MLE} = \frac{x_j}{\sum_i x_i}$$

The data above will give $\theta_1^{MLE} = 0.0200$, $\theta_{20}^{MLE} = 0.0159$ and $\theta_{38}^{MLE} = 0.0255$. Contrast this with the MAP solution, the conjugate prior of a multinomial distribution is the Dirichlet distribution:

$$D(\theta; \mathbf{c}) = \frac{\Gamma(\sum_i c_i)}{\prod_i \Gamma(c_i)} \prod_i \theta_i^{c_i - 1} \tag{2.23}$$

In MAP estimation the use of a conjugate distribution is not essential, but as it is often convenient to do so in more advanced Monte Carlo methods, to remain consistent we shall use it here. The c_i parameters in the Dirichlet density 2.23 are interpreted as a *prior count*, a prior belief that ball i will have been drawn c_i times. If we assume the balls are unbiased, over all 1006 draws of the lottery each ball would expect to have appeared a little over 123 times. Thus we can set $c_i = 123$ for all i. Note, the Dirichlet distribution does not require an integer parameter but we shall use one here for convenience.

Using equation 2.1.6 the log-posterior becomes:

$$\log L(\Theta|Data) \propto \log(49!) - \sum_{i=1}^{49} \log(x_i!) + \sum_{i=1}^{49} x_i \log(\theta_i) + \sum_{i=1}^{49} (123-1)\log(\theta_i) \qquad (2.24)$$

and hence the MAP solution is:

$$\theta_j^{MAP} = \frac{x_j + 122}{\sum_i x_i + 122}$$

The data above will give $\theta_1^{MAP} = 0.0202$, $\theta_{20}^{MAP} = 0.0182$ and $\theta_{38}^{MAP} = 0.0230$. These multinomial parameters are now close to the unbiased value of 0.0204. It may appear that this is because we have in fact cheated as we in a sense knew the answer we were looking for before we started. But the use of priors over the parameters is important in a number of ways. It is rarely the case that we know nothing about the problem in hand, this knowledge can be incorporated into the prior. A common consequence of prior distributions is a smoothing of model parameters, they can also be used to avoid over and under-fitting. Finally without prior distributions on the model parameters a method cannot claim to be Bayesian, as the fundamental Bayesian assumption is that model parameters are not fixed but are drawn from a distribution.

2.2 Graphical Models

In this section we shall first introduce the idea of conditional independence, and how this is related to the joint distribution of a number of variables. If two variables are independent then simply $P(A, B) = P(A)P(B)$, that is the joint distribution of two variables A and B can be decomposed into separate distributions for A and B. This means for independent variables, A and B, conditioning one variable on a particular value of the other has no effect, hence $P(A|B) = P(A)$. Intuitively speaking, knowing the value of B does not alter our belief about the values of A. Conditional independence is an extension of this. If two random

variables A and B are conditionally independent given a random variable Z, then

$$P(A, B|Z) = P(A|Z)P(B|Z)$$

similarly

$$P(A|B, Z) = P(A|Z)$$

Intuitively this is saying that once we know the value of Z then knowing B adds nothing more to our belief about the values of A. It is an important point to make that conditional independence between A and B given Z does not imply that $P(A|B) = P(A)$. In probabilistic modelling we are often interested in modelling the joint distribution $P(X_1, \ldots, X_n)$ of a set of random variables $\{X_1, \ldots X_n\}$. If we know the conditional dependencies between the variables X_i then we can write the joint distribution in a factored form which will simplify any computations.

Graphical Models are a way of representing the conditional dependencies between random variables in a probabilistic model. In all the graphical models we give in this thesis the random variables are represented by nodes with conditional dependencies depicted as arcs between these nodes. Circular nodes indicate that we assume a full distribution for this variable, while square nodes indicate a point estimate. Shaded nodes represent an observed variable while unshaded nodes indicate the need to perform inference for that node. Additionally frames will denote exchangeability in a sequence of random variables.

Thus the simplest non trivial (the most trivial being a single variable) graphical model possible would be that given in figure 2.1, which has the corresponding equation 2.25. An example of a model with exchangeability is given by figure 2.2 and equation 2.26. Figure 2.3 gives a simple practical graphical model, this is model for calculating the average (the mean here is taken as a point value) of N observed variables X_n. Figure 2.4 shows how graphical models can be used to express the conditional independence between variables and how this then related the joint distribution to a factorised form.

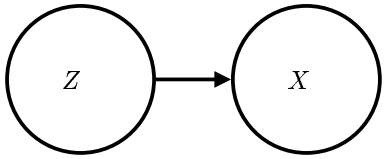


Fig. 2.1: Graphical Model for equation 2.25

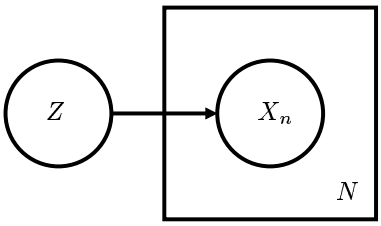


Fig. 2.2: Graphical Model for equation 2.26

$$P(X, Z) = P(X|Z)P(Z) \tag{2.25}$$

$$P(X, Z) = \prod_{n}^{N} P(Z)P(\mathbf{X}_n|Z) \tag{2.26}$$

Bayesian networks are examples of graphical models, they are directed acyclic graphs (DAGs). In these the conditional dependencies indicated by the graph structure show how to factorise the joint probability distribution over all the variables into a simpler form. This allows tractable calculations of conditional distributions.

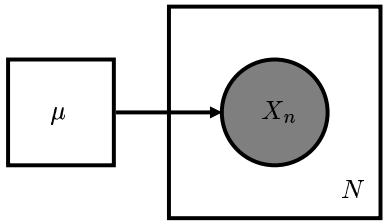


Fig. 2.3: Graphical Model for estimating the mean of a variable from a set of observed values

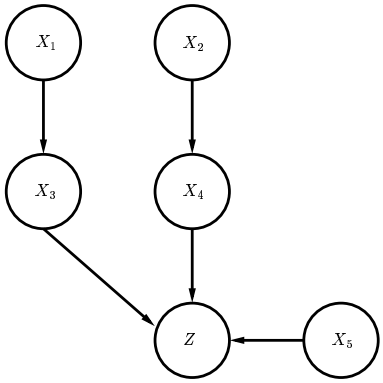


Fig. 2.4: This graphical model demonstrates the relationship between the joint density and its factorised form. In particular it draws attention to the the conditional independence between Z and X_1 and X_2. Namely $P(Z, X_1, X_2, X_3, X_4, X_5) = P(Z|X_3, X_4, X_5)P(X_3|X_1)P(X_4|X_2)P(X_1)P(X_2)P(X_5)$

2.2.1 Mixture Models

Here we shall introduce the graphical models for a Gaussian Mixture model and for the Latent Dirichlet Allocation model ([15]).

Gaussian Mixture

Let us define our data set as a number of observations $\mathbf{X}_d$ (d is an index for observations), with each observation having a number of features X_{df} (f is an index for features). The assumption behind all mixture models is that there is no single distribution (by that we mean set of parameters rather than class of distribution) that generated, $\mathbf{X}_{\cdot f}$, a particular feature across the whole data set, rather a set of distributions. We shall index this set $k = 1, \ldots, K$. The proportion of observations that were generated by each of K parameter choices is known as the mixing parameter.

The graphical model for a Gaussian Mixture is given in figure 2.5. The likelihood of generating a sequence of samples, $\mathbf{X}_1, \ldots, \mathbf{X}_D$, from this model is written is equation 2.27. In a Gaussian Mixture model the mixture is selected once for each observation, that is the parameters defining each feature for that observation are all conditioned on the same k.

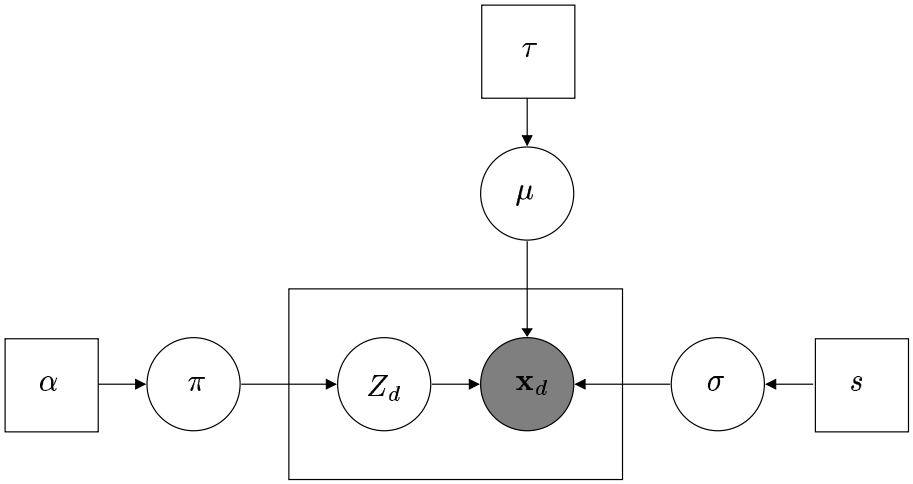


Fig. 2.5: Graphical Model for a Gaussian Mixture

$$\left[\prod_{d}^{D}\sum_{k} P(Z_d = k|\boldsymbol{\pi})(\mathbf{X}_d|\mu_k, \sigma_k^2)\right] P(\boldsymbol{\mu}|\tau)P(\boldsymbol{\sigma}^2|s)P(\boldsymbol{\pi}|\alpha) \tag{2.27}$$

Where for multidimensional data with F features $P(\mathbf{X}_d) = \prod_f P(X_{df})$

It is sometimes useful to write the generative process in words, here $a \sim P(a|b)$ means we sample a from a distribution with parameters b and a boldface $\boldsymbol{a}$ indicates a vector of parameters was sampled:

Sample $\boldsymbol{\pi} \sim P(\boldsymbol{\pi}|\alpha)$
Sample $\boldsymbol{\mu} \sim P(\boldsymbol{\mu}|\tau)$
Sample $\boldsymbol{\sigma}^2 \sim P(\boldsymbol{\sigma}^2|s)$
while $d \leq D$ **do**
  Sample $Z_{df} \sim Mult(\boldsymbol{\pi})$
  while $f \leq F$ **do**
    Sample $X_{df} \sim \mathcal{N}(\mu_{Z_{df}}, \sigma^2_{Z_{df}})$
  end
end

Algorithm 1: Generative Process for a Gaussian Mixture. This is a fully Bayesian treatment of a Gaussian Mixture as all model parameters are themselves random variables

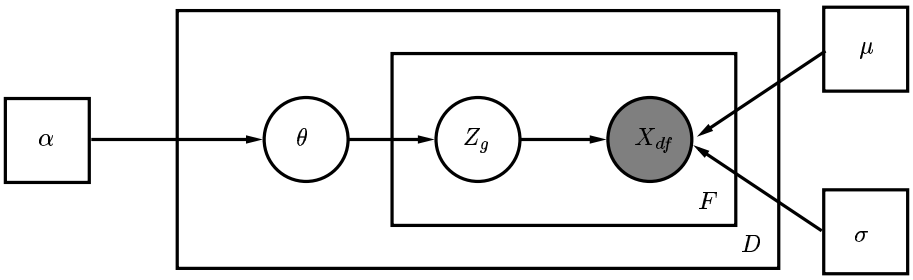


Fig. 2.6: Graphical Model for Latent Process Decomposition

The Latent Dirichlet Allocation [15] model is similar to that of a mixture model. It was first introduced as a generative model for textual data in which there was more than one distinct theme running through the text. Because of this it is often referred to as an *aspect* mode. It offers more flexibility than a standard mixture model. For each observation a multinomial across mixtures is generated by sampling a Dirichlet distribution. Then for each feature a mixture is sampled from this multinomial. Thus for a given observation the features X_{df} need not have been generated by the same mixture. In Rogers *et al*, [70], they applied LDA to continuous data but substituted a Gaussian distribution for the original Multinomial. The likelihood of the resulting Latent Process Decomposition (LPD) model is given in equation 2.28, with corresponding graphical model 2.6.

$$\prod_d \left[\prod_f \sum_k P(Z_{df} = k|\boldsymbol{\theta}_d) P(X_{df}|\mu_{Z_d f}, \sigma^2_{Z_d f}) \right] P(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) \tag{2.28}$$

Latent Process Decomposition

The generative process for LPD is given as.

while $d \leq D$ **do**
  Sample $\boldsymbol{\theta} \sim Dirichlet(\boldsymbol{\alpha})$
  while $f \leq F$ **do**
    Sample $Z_{df} \sim Mult(\boldsymbol{\theta})$
    Sample $X_{df} \sim \mathcal{N}(\mu_{Z_{df}}, \sigma^2_{Z_{df}})$
  end
end

Algorithm 2: Generative Process for a Latent Process Decomposition. In this example the construction is not a fully Bayesian treatment of LPD. The model parameters are not drawn from a distribution and so are not random variables.

2.2.2 Biomedical Relevance of Mixture Models

Here we shall justify the application of decomposable models, such as the **Gaussian Mixture** and **Latent Process Decomposition**, to biomedical data. We shall also show the differences between the **Gaussian Mixture** and **Latent Process Decomposition** models.

Data sets containing more than one group of individuals, or more than one distinct class elements occur frequently in the biomedical sciences. For example, figure 2.7 is a histogram of pixel intensity for a single CT scan. It is clear that there are a number of distinct groups present in the data appearing at different positions on the spectrum of intensity. Additionally in this particular example, each of the distinct groups seems to have a spread and overall frequency (compared to the other groupings) associate with it. The figure 2.8 is a histogram of the gene expression for a single gene across a number of patients. Again, it is clear from this there appear to be a number of distinct groups in the data.

If we assume normally distributed data (this is generally a valid assumption on data that occurs naturally) and there existed a single group the histogram would look something like that given in figure 2.9. If we extend this to a situation where there are three grouping in the data, each with a distinct mean. The histogram of this new dataset would look something figure 2.10, additionally figure 2.11 shows the separate components making up the dataset. Although this is just a visual example we can already see that the mixture distributions given figures 2.8 and 2.10 are beginning to look very similar, it is easy to accept that with a suitable choice of parameters a mixture histogram could be generated to look like the CT data given

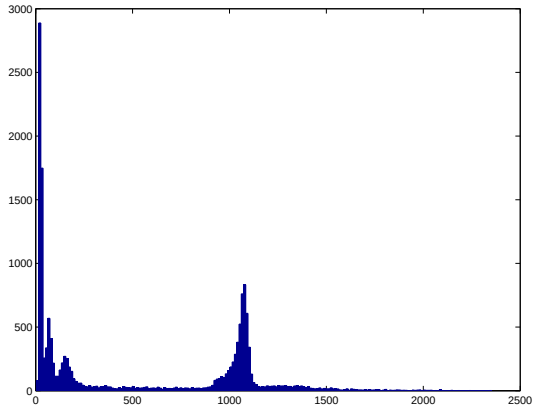


Fig. 2.7: A histogram of pixel intensity for a single CT scan

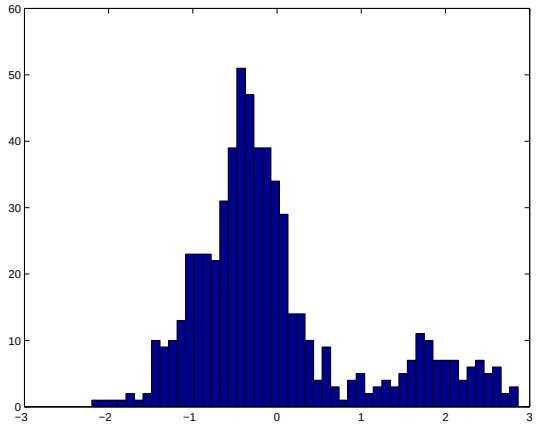


Fig. 2.8: A histogram of gene expression for a single gene

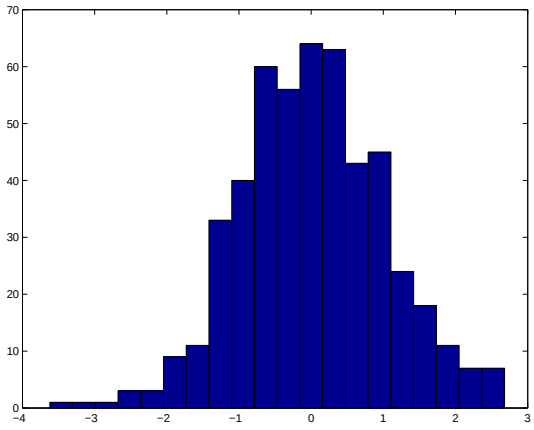


Fig. 2.9: A histogram of samples from a zero mean unit variance Gaussian.

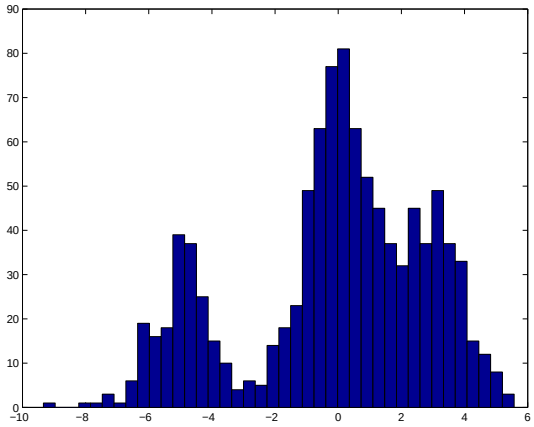


Fig. 2.10: A histogram of a data set containing samples from three Gaussian distributions. Each has unit variance, with means -5, 0 and 3.

in figure 2.7.

In the situation we have demonstrated above a **Gaussian Mixture** is a good way of modelling the data from figures 2.7 and 2.8. In the graphical model given in figure 2.5 we have three model parameters. These are the means, μ which dictate the centres of the different groupings, the variance σ^2 which dictate the spread of each group and the mixing parameter π which dictates the overall frequency of each group.

So far we have only considered simple examples, in figure 2.7 we have a single ct scan from a single patient and in figure 2.8 we have a single gene from a cohort of patients. In a more

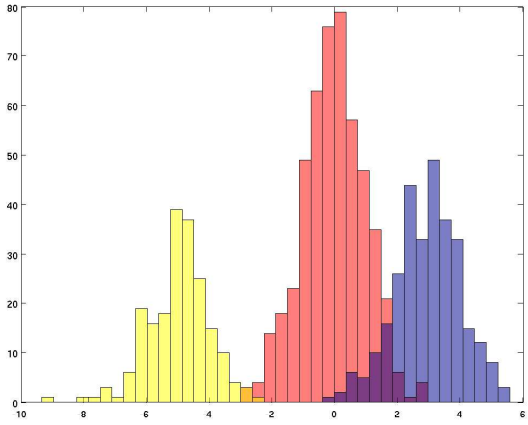


Fig. 2.11: A histogram of the same data give in figure 2.10. This shows the separate components that make up the whole dataset.

realistic situation we would have multiple scans from more than one patient and a number of genes taken across the whole patient cohort.

In the case of gene expression, each sample is a patient with a number of expression $\boldsymbol{E}_d = \{E_{d1}, \ldots, E_{dG}\}$. We can easily extend a Gaussian mixture to a multidimensional case. The model parameters μ and σ^2 are now extended to have a second dimension indexed by gene g, in the case where we assume an isotropic covariance (this is the assumption of uncorrelated variables and unrealistic in general), this is in effect modelling each gene in a similar way to that as demonstrated by figure 2.8. However the mixing parameter π remains of dimension $1 \times K$ where k the number of mixtures chosen. Because of this, the assumption is for a single patient that a single value for K is chosen and this is then used across all that patients genes. Each gene still has a distinct set of means associated with it $\{\mu_{g1}, \ldots, \mu_{gK}\}$ for each choice of k, but for each patient in the group all the genes are tied to the particular chosen value of parameter k, $\{\mu_{1k}, \ldots, \mu_{Gk}\}$. This is quite restrictive, it does allow different groups within the dataset but it also assumes that for all patients within a group any given gene across the group is distributed according to a single Gaussian. Figure 2.12 is a simple schematic representation of this idea. In this example there are 3 possible mixture choice, red, green and blue represented by a *bag of balls*. With $p(Green) = 0.6$ and $p(Red) = p(Blue) = 0.2$. For each patient, represented by a column, a single ball is chosen at random - this corresponds to the value of k. The parameters for each gene μ_{gk} and σ_{gk} are then used to generate each gene expression. Compare this to the schematic representation of the LPD model, give in

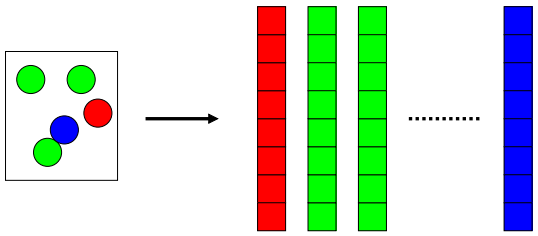


Fig. 2.12: Generative Process of a Mixture Model. The box of spheres represents a multinomial distribution and there is a single column for each patient with 8 genes each represented by a single coloured box.

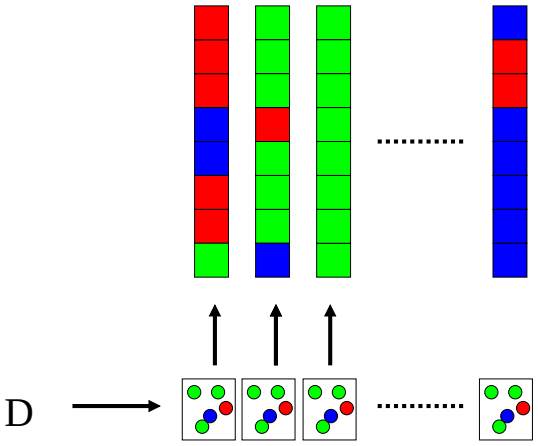


Fig. 2.13: Generative Process of a an LPD mixture model. The **D** represents a Dirichlet distribution. Each box of spheres represents a multinomial distribution specific to each patient. Again there is a single column for each patient with 8 genes. Note that although the graphic is the same each box of spheres represents a separate draw from a Dirichlet distribution.

figure 2.13. In this model for each patient a multinomial (represented again by the bag of balls) is drawn from a Dirichlet distribution (represented by D in figure 2.13), the process k is then drawn from this specific multinomial for each gene. Hence it is now possible to have a mixture of processes in an individual patient (this is seen as a mixture of coloured boxes in a specific column). The gene expressions are still *tied* together as their processes are drawn from the same multinomial, and the multinomial's are *tied* as they are drawn from the same Dirichlet.

In this thesis the motivation for using LPD based mixture models on gene expression data is that we weakly associate a process (that is a distinct set of parameter values) with a distinct biological process. The connection we make is that there may exist a finite number

of distinct underlying genetic effects (the total number of processes k) , these directly affect the expression for a group of genes. Each patient in the cohort is affected by these effects in varying degrees, this is analogous to the patient specific multinomial. In the graphical model for LPD given in 2.6 there are two levels exchangeability, indexed by D and F. With reference to gene expression for a number of patients over a number of genes, exchangeability in D indicates that we allow any permutation of patients without any impact. Similarly exchangeability in F indicates we can have a global permutation of genes without any effect, indeed in the context of gene expression a specific ordering of the genes is meaningless. Although only a great simplification of the biology the LPD model may provide an interesting picture of gene expression.

Turning out attention to CT images, let us first restrict ourselves to the case when we have just a single image for each patient and also restrict ourselves to the case where we are only considering the intensity of the pixel (later we shall use more advances image features). Recall that figure 2.7 is a histogram of pixel intensities for a single CT image from a single patient. It is clear from this that there exist distinct groupings within a single image. So for each patient we have a mixture over distinct Gaussians. In the same way we weakly associated different mixture components with biological processes for gene expression we attempt to associate different mixture components with distinct tissue types. Thus, 2.7 can be viewed as a decomposition of a patient into distinct image appearances which are closely associated with normal tissue and different disease types. Indeed, this is a very valid assumption as *Emphysema* is sometimes defined purely in terms of its pixel intensity. To model the CT data using a standard mixture of Gaussians, each image would only be allowed to contain a single Gaussian. This is obviously far too restrictive, but in the case of an LDA based mixture model each patient (that is each image) is a distinct mixture over all the Gaussians available. The proportion of each corresponding to the proportion of each tissue type present. Once again this is a simplification of the underlying biology, but it is certainly expressive enough to allow an interesting analysis of the CT data.

2.3 Methods of Inference

2.3.1 Expectation Maximisation

One very important development in the field of probabilistic inference was the Expectation Maximisation algorithm [26]; a good introduction is given here [13]. The main idea behind the EM algorithm is to decompose likelihoods in terms of *Observed Data* and *Missing Data*. Using this construction we can alternate between approximating the missing data and optimising the model parameters. We shall now give a brief overview of the EM algorithm.

Overview of traditional EM

Let us start by denoting *Observed data* by X and *Missing data* by Z. X and Z both denote sets of variables. As examples: Z could be the unknown class label in a clustering algorithm or a missing gene expression from some microarray data, X could be a vector of features for a given sample. Let θ denote the parameters of the model. As EM is an iterative procedure denote θ_n to be an estimate of θ after n iterations.

We denote

$$p(X|\theta) \tag{2.29}$$

as the *incomplete data* likelihood and

$$p(X, Z|\theta) \tag{2.30}$$

as the *completed data* likelihood.

Since

$$p(Z, X|\theta_n) = p(X|\theta_n)P(Z|X, \theta_n)$$

We can then write the *incomplete data* likelihood in terms of a *completed data* likelihood and a marginal probability of the missing data.

$$p(X|\theta_n) = \frac{p(Z, X|\theta_n)}{P(Z|X, \theta_n)} \tag{2.31}$$

For simplicity we shall work with log-likelihoods. As log is monotonically increasing all derived inequalities will hold for non-logged functions. Expanding the *incomplete data* log-likelihood and utilising equation (2.31).

$$\begin{aligned}
\log(p(X|\theta_n)) &= \textstyle\sum_Z P(Z|X, \theta_n) \log(p(X|\theta_n)) \\
&= \textstyle\sum_Z P(Z|X, \theta_n) \log\left[\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}\right] \\
&= \textstyle\sum_Z P(Z|X, \theta_n) \log p(Z, X|\theta_n) - \sum_Z P(Z|X, \theta_n) \log P(Z|X, \theta_n) \\
&= E_Z\left[\log p(Z, X|\theta_n)|X, \theta_n\right] - E_Z\left[\log p(Z|X, \theta_n)|X, \theta_n\right] \\
&= Q(\theta_n|\theta_n) + R(\theta_n|\theta_n)
\end{aligned} \tag{2.32}$$

We have decomposed the *incomplete data* log-likelihood into two terms. The first of these, Q, is the expectation of the *completed data* log-likelihood with respect to current estimate of the *missing data*. The second term, R, is the entropy of the current estimate of the *missing data*. The use of Q to denote this expectation is taken from [26]. Note we have simply expressed the *incomplete data* log-likelihood in terms of the KL divergence given in equation 2.19, with $p = P(Z|X, \theta_n)$ and $q = p(Z, X|\theta_n)$. In the standard convention we define:

$$Q(\theta|\theta_n) \equiv E_Z\left[\log p(Z, X|\theta)|X, \theta_n\right] \tag{2.33}$$

and

$$R(\theta|\theta_n) \equiv -E_Z \left[\log p(Z|X,\theta)|X,\theta_n\right] \tag{2.34}$$

The basic idea of the EM algorithm is to construct an iterative scheme which, at each iteration, increases the *incomplete data* likelihood. The following is a very standard approach for proof of convergence of EM.

$$\begin{aligned}
\log(p(X|\theta_n)) &= \log\left[\textstyle\sum_Z p(Z,X|\theta_n)\right] \\
&= \log\left[\textstyle\sum_Z P(Z|X,\theta_n)\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}\right] \\
&= \log\left[E_Z\left[\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}|X,\theta_n\right]\right]
\end{aligned} \tag{2.35}$$

This is the expectation of $\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}$, with respect to the missing data Z, given the observed data X and the estimate of the parameters after n iterations θ_n. Using the form Jensen's inequality give in equation 2.17:

$$\begin{aligned}
\log\left[E_Z\left[\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}|X,\theta_n\right]\right] &\geq E_Z\left[\log\left[\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}|X,\theta_n\right]\right] \\
&= \textstyle\sum_Z P(Z|X,\theta_n)\left[\log\frac{p(Z,X|\theta_n)}{P(Z|X,\theta_n)}\right] \\
&= \textstyle\sum_Z P(Z|X,\theta_n)\left[\log p(Z,X|\theta_n) - \log P(Z|X,\theta_n)\right] \\
&= E_Z\left[\log p(Z,X|\theta)|X,\theta_n\right] - E_Z\left[\log p(Z|X,\theta)|X,\theta_n\right] \\
&= Q(\theta|\theta_n) + R(\theta|\theta_n)
\end{aligned} \tag{2.36}$$

Thus we have established the following inequality.

$$\log(p(X|\theta_n)) \geq Q(\theta|\theta_n) + R(\theta|\theta_n)$$

We can find a new parameter estimate θ_{n+1} such that $\theta_{n+1} = argmax_\theta Q(\theta|\theta_n)$, hence

$$Q(\theta_{n+1}|\theta_n) \geq Q(\theta_n|\theta_n) \tag{2.37}$$

and adding $R(\theta_n|\theta_n)$ to equation 2.37 and using equation 2.32

$$\begin{aligned} L(X|\theta_{n+1}) &\geq Q(\theta_{n+1}|\theta_n) + R(\theta_n|\theta_n) \\ &\geq Q(\theta_n|\theta_n) + R(\theta_n|\theta_n) \\ &= L(X|\theta_{n+1}) \end{aligned} \tag{2.38}$$

hence

$$L(X|\theta_{n+1}) \geq L(X|\theta_n)$$

So for each selection of θ_{n+1} an increase in Q corresponds to an increase in the true, *incomplete data*, likelihood. The EM algorithm can be summarised as follows:

- E-Step: Calculate $Q(\theta|\theta_n)$
- M-Step: Find $\theta_{n+1} = argmax_\theta Q(\theta|\theta_n)$

Variants on EM

One of the great limitations of the EM algorithm is in its convergence properties. The EM algorithm will only converge to local minima, in fact it will only converge to a point estimate of the mode of a bound on the posterior. This in itself is often sufficient to give a good model for some data, but compared for example to Monte Carlo techniques (see section 2.3.7) where the full posterior is obtained a point estimate is inferior. This however is balanced by ease of implementation, and particularly speed of computation. The second problem that is well known is that the EM algorithm has a slow convergence rate. This can be due to over fitting of the latent states (E-Step) and is in part remedied by a series of incomplete E and M steps. A full discussion and application to Mixture models is given in [23].

2.3.2 Variational Inference

In many ways variational inference is a generalisation of the EM algorithm. An excellent and thorough introduction and overview to their usage can be found in [85]. Variational methods are used when exact inference within a graphical model is not possible. Via the introduction of variational distributions a bound on original probabilities is constructed. The motivation being that inference within this bound is often easy to perform. There are two distinct approaches to Variational Inference. The first is most similar in motivation to the standard EM algorithm and provides an iterative procedure to generate ML or MAP point estimates.

As a simple example we shall show how to perform maximum likelihood inference in a simple mixture model. Suppose we have a mixture model with logged likelihood:

$$\log P(X|\Theta) = \log \sum_k \pi_k P(X|\Theta, k) \tag{2.39}$$

We shall introduce the discrete variational distribution γ.

$$\begin{aligned} \log P(X|\Theta) &= \log \textstyle\sum_k \pi_k \frac{\gamma_k}{\gamma_k} P(X|\Theta, k) \\ &\geq \textstyle\sum_k \gamma_k \log \pi_k \frac{P(X|\Theta,k)}{\gamma_k} \end{aligned} \tag{2.40}$$

The bound is introduced by using Jensen's inequality from equation 2.17 as the summation over k is actually an expectation over the variational distribution. Equation 2.40 is now a tractable bound. This bounded likelihood can now be maximised by iteratively updating the maximum likelihood solution of the model parameters and the maximum likelihood solution of the variational distribution until convergence. As is the case with the EM algorithm, variational inference will give a point estimate of the posterior distribution.

In keeping with the Bayesian methodology the second approach to Variational Inference attempts to *explain away* any latent uncertainty in our model by integrating out all hidden nodes. The is formulated by constructing a bound based on the negative free energy of the system and maximising this. Among the first authors to adopt this approach was Attias in [7]. Here we shall give a general derivation of the *Variational Bayes* algorithm and a example of it in application to **Latent Process Decomposition**.

One advantage of a Variational Bayesian approach to parameter estimation over an EM algorithm is that model comparison can be performed more easily. In an EM algorithm a cross validation is required. This involves retaining a certain percentage of the data and then estimating the parameters on the remaining data. The likelihood of the left out data is then calculated as a score for the accuracy of the model. Over-fitted models are penalised as the score is based on the retained data not used in parameter estimation. In the case of models containing Mixtures of distributions a cross validation can be performed for varying numbers of mixtures and then the scores for each compared to give a best fit model. One criticism of this approach is that it is not strictly correct to directly the likelihoods of different models. Integrating the likelihood function of some data, given the model parameters and the number of mixtures k, with respect to the parameters will give one.

$$p(\boldsymbol{D}|K) = \int \mathcal{L}(\boldsymbol{D}|\boldsymbol{\Theta}, K) d\boldsymbol{\Theta} = 1$$

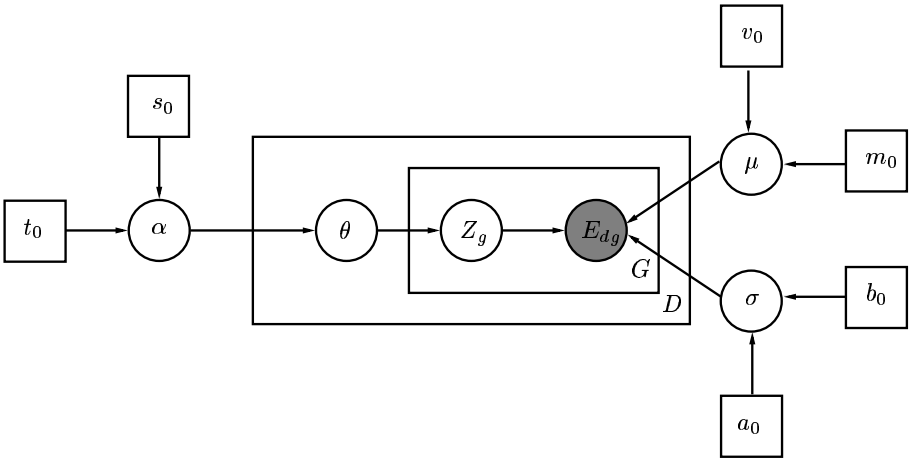


Fig. 2.14: Graphical Model for a fully Bayesian Latent Process Decomposition

Therefore, parameter rich models will be given a higher likelihood than others resulting in over fitting in a model sense when estimating K through likelihood comparison. Conversely in a Variational Bayesian all model parameters are integrated out so it is possible to give a lower bound on $p(\boldsymbol{D}|K)$, and hence direct model comparison is valid.

2.3.3 Variational Bayesian Inference

Here we present the general methodology for Variational Bayesian Inference (VB). VB seeks to find a lower bound on the evidence $p(Data)$ which is in a tractable form to be maximised. Approximations are made to the posterior distributions of all hidden and model variables so that they can be marginalised (integrated out). At each iteration of VB it is the hyper-parameters, rather than parameters, that are updated. Thus compared to the EM algorithm, or a variational approach that utilises Jensen's inequality the emphasis is shifted a step up-wards.

In the derivation there are two hidden nodes $\boldsymbol{Z}$ and $\boldsymbol{\theta}$ and a set of parameters $\boldsymbol{\Theta}$. The reason for this explicit selection of the variables is they correspond the the variables in the fully Bayesian LPD model given in figure 2.14 which we shall use as an example.

In summary, the variables in our model are:

- E_{dg} Expression for gene.
- $\boldsymbol{\theta}, \boldsymbol{Z}$ Hidden variables
- $\boldsymbol{\Theta}$ model parameters (note these are not hyper parameters, but the distributions governed by the hyper-parameters)
- All square boxed variables are hyper-parameters for which we estimate a point value.

2.3.4 General VB

The evidence of some data $p(E)$ can be written as a ratio of the joint distribution (with respect to some variables) $p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z})$ and the posterior distribution of these variables given the data $p(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})$.

$$p(\boldsymbol{E}) = \frac{p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z})}{p(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})} \tag{2.41}$$

The log of this is written as:

$$\log p(\boldsymbol{E}) = \log p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}) - \log p(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}) \tag{2.42}$$

Let us introduce an approximation to the posterior distributions of all model and hidden variables $q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})$. If we take expectations of expression 2.42 with respect to this approximate posterior $q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})$, the left hand side remains unchanged as this is independent of $\boldsymbol{\Theta}$, $\boldsymbol{\theta}$ and $\boldsymbol{Z}$.

$$\log p(\boldsymbol{E}) = \int q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}) \log p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}) d\boldsymbol{\Theta} d\boldsymbol{\theta} d\boldsymbol{Z} - \int q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}) \log p(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}) d\boldsymbol{\Theta} d\boldsymbol{\theta} d\boldsymbol{Z}$$

Multiplying $p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z})$ top and bottom by $q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})$ and separating the terms we can

now write

$$\log p(\boldsymbol{E}) = F(\boldsymbol{\Theta}) + KL(q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})||p(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}))$$

where

$$F(\boldsymbol{\Theta}) = \int q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}) \log \frac{p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z})}{q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E})} d\boldsymbol{\Theta} d\boldsymbol{\theta} d\boldsymbol{Z}$$

As the KL divergence is strictly greater than zero, we can now say that

$$\log p(\boldsymbol{E}) \geq F(\boldsymbol{\Theta})$$

Equality holds when $KL = 0$, this is true when the approximate posterior q and true posterior p coincide, the case when our approximation becomes exact. The idea behind a **Variational Bayes** approach is to maximise the evidence by maximising $F(\boldsymbol{\Theta})$.

We shall now make an important assumption about the posterior. We assume that it factorises into separate terms, such that $q(\boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{E}) = q(\boldsymbol{\Theta})q(\boldsymbol{\theta})q(\boldsymbol{Z})$ where the dependence on $\boldsymbol{E}$ is implied.

By writing $p(\boldsymbol{E}, \boldsymbol{\Theta}, \boldsymbol{\theta}, \boldsymbol{Z}) = p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta})p(\boldsymbol{\Theta})$ we can now expand $F(\boldsymbol{\Theta})$ as

$$F(\boldsymbol{\Theta}) = \int q(\boldsymbol{\Theta})q(\boldsymbol{\theta})q(\boldsymbol{Z}) \log \frac{p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta})p(\boldsymbol{\Theta})}{q(\boldsymbol{\Theta})q(\boldsymbol{\theta})q(\boldsymbol{Z})} d\boldsymbol{\Theta} d\boldsymbol{\theta} d\boldsymbol{Z}$$

Thus by expanding and integrating out $q(\boldsymbol{\theta})$ and $q(\boldsymbol{Z})$

$$F(\boldsymbol{\Theta}) = \int q(\boldsymbol{\Theta})q(\boldsymbol{\theta})q(\boldsymbol{Z}) \log \frac{p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta})}{q(\boldsymbol{\theta})q(\boldsymbol{Z})} d\boldsymbol{\Theta} d\boldsymbol{\theta} d\boldsymbol{Z} - KL(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta})) \tag{2.43}$$

In equation 2.43 the first term is an averaged likelihood and the second term $-KL(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta}))$ is a measure of the *distance* between approximate posterior and prior over parameters, as this term increases with the number of parameters it can be seen as a penalising term for over complex models. Indeed it has been shown that in certain situations this reduces to the Bayesian information criteria (BIC) and the Minimum Description Length (MDL) (see [7] for further details).

To maximise $F(\boldsymbol{\Theta})$ in equation 2.43 we take zeroed gradients (functional derivatives in this case) with respect to the approximate posteriors $q(\boldsymbol{\Theta})$, $q(\boldsymbol{\theta})$ and $q(\boldsymbol{Z})$.

$$\frac{\delta F(\boldsymbol{\Theta})}{\delta q(\boldsymbol{\theta})} = \int q(\boldsymbol{\Theta})q(\boldsymbol{Z}) \log \frac{p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta})}{q(\boldsymbol{\theta})q(\boldsymbol{Z})} d\boldsymbol{\Theta} d\boldsymbol{Z} - \int \frac{q(\boldsymbol{\Theta})q(\boldsymbol{\theta})q(\boldsymbol{Z})}{q(\boldsymbol{\theta})} d\boldsymbol{\Theta} d\boldsymbol{Z} = 0$$

$$\int q(\boldsymbol{\Theta})q(\boldsymbol{Z}) \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) d\boldsymbol{\Theta} d\boldsymbol{Z} - 1 - \log q(\boldsymbol{\theta}) \int q(\boldsymbol{\Theta})q(\boldsymbol{Z}) d\boldsymbol{\Theta} d\boldsymbol{Z} - \int q(\boldsymbol{\Theta})q(\boldsymbol{Z}) \log q(\boldsymbol{Z}) d\boldsymbol{\Theta} d\boldsymbol{Z} = 0$$

As the densities $q(\dots)$ integrate to one we can write

$$q(\boldsymbol{\theta}) \propto \exp \left[\int q(\boldsymbol{\Theta})q(\boldsymbol{Z}) \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) d\boldsymbol{\Theta} d\boldsymbol{Z} \right] \tag{2.44}$$

Analogously

$$q(\boldsymbol{Z}) \propto \exp \left[\int q(\boldsymbol{\Theta})q(\boldsymbol{\theta}) \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) d\boldsymbol{\Theta} d\boldsymbol{\theta} \right] \tag{2.45}$$

For any of the model parameters,

$$\frac{\delta F(\boldsymbol{\Theta})}{\delta q(\boldsymbol{\Theta})} = \int q(\boldsymbol{\theta})q(\boldsymbol{Z}) \log \frac{p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta})}{q(\boldsymbol{\theta})q(\boldsymbol{Z})} d\boldsymbol{\theta} d\boldsymbol{Z} - \log \frac{p(\boldsymbol{\Theta})}{q(\boldsymbol{\Theta})} - 1 = 0$$

so

$$q(\boldsymbol{\Theta}) \propto \exp\left[\int q(\boldsymbol{\theta})q(\boldsymbol{Z}) \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) d\boldsymbol{\theta} d\boldsymbol{Z}\right] p(\boldsymbol{\Theta}) \tag{2.46}$$

Equations 2.44 to 2.46 give the approximate posterior distributions for the latent variables and model parameters. They can be interpreted as the posterior taking the form of the exponential of the averaged log likelihood over all remaining variables. Thus all uncertainty is integrated away. The posterior forms of $q(\boldsymbol{\Theta})$, $q(\boldsymbol{\theta})$ and $q(\boldsymbol{Z})$ are determined directly from the optimisation via equations 2.44 to 2.46. In the case of model parameters, the prior distributions in 2.46 are chosen as conjugate to the derived exponentials so that the parametric form for $q(\boldsymbol{\Theta})$ remains the same.

Having derived the general form of the posterior distributions in a VB approach we shall use the LPD model given in figure 2.14 as an example.

2.3.5 Application to LPD

The derivation acts as an extension to that given for mixture models in [7]. As the prior distributions are chosen to maintain functional form there is a certain amount of *hindsight* used in the presentation.

The joint likelihood of the observed data $\boldsymbol{E}$ and the latent variables $\boldsymbol{\theta}, \boldsymbol{Z}$, for the model given graphically in figure 2.14, is written.

$$P(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) = \prod_d p(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) \prod_g p(Z_{dg}|\boldsymbol{\theta}_d) p(E_{dg}|\mu_g, \beta_g, Z_{dg}|\boldsymbol{\theta}_d) \tag{2.47}$$

By extending Z_{dg} to a k dimensional vector of zeros with a 1 in the location of the original Z_{dg} this can be re-expressed as

$$P(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) = \prod_d p(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) \prod_{g,k} [p(Z_{dg,k}|\boldsymbol{\theta}_d) p(E_{dg}|\mu_g, \beta_g, Z_{dg,k})]^{Z_{dg,k}}$$

Thus the log joint likelihood is written in equation 2.48.

$$\begin{aligned} \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) &= \textstyle\sum_{d,k} (\alpha_k - 1) \log \theta_{dk} \\ &+ \textstyle\sum_{d,g,k} Z_{dg,k} \left[\log \theta_{dk} - 0.5 \beta_{gk} (E_{dg} - \mu_{gk})^2 + 0.5 \log \beta_{gk} \right] \end{aligned} \tag{2.48}$$

We endow the model parameters with prior distributions, and give the form of the distributions for the latent variables as

- $p(\boldsymbol{\alpha}) = \prod_k p(\alpha_k) \sim \prod_k \Gamma(\alpha_k; t_0, s_0)$
- $p(\boldsymbol{\mu}) \sim \prod_{gk} \mathcal{N}(\mu_{gk}; m_0, v_0)$
- $p(\boldsymbol{\beta}) \sim \prod_{gk} \Gamma(\beta_{gk}; a_0, b_0)$
- $p(\boldsymbol{Z}|\boldsymbol{\theta}) \sim \prod_{dgk} \theta_{d,k}^{Z_{dg,k}}$
- $p(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \prod_d p(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) \sim Dirichlet(\boldsymbol{\alpha})$

We need to take expectations of the log likelihood given in equation 2.48 with respect to the approximate posterior distributions $q(\boldsymbol{\theta})$, $q(\boldsymbol{\alpha})$, $q(\boldsymbol{\mu})$ and $q(\boldsymbol{\beta})$. The approximate posteriors are assumed to factorise and have the form

- $q(\boldsymbol{\theta}) = \prod_d q(\boldsymbol{\theta}_d) \sim \prod_d Dirichlet(\tilde{\boldsymbol{\alpha}}_d)$
- $q(\boldsymbol{\alpha}) = \prod_k q(\alpha_k) \sim \prod_k \Gamma(t_k, s_k)$
- $q(\boldsymbol{\mu}) \sim \prod_{gk} \mathcal{N}(\tilde{m}_{gk}, \tilde{v}_{gk})$
- $q(\boldsymbol{\beta}) \sim \prod_{gk} \Gamma(\tilde{a}_{gk}, \tilde{b}_{gk})$
- $q(\boldsymbol{Z}) \sim \prod_{dgk} r_{dg,k}^{Z_{dg,k}}$

Leaving out the parameter of interest and taking expectations with respect the the posterior distributions $q()$ of all the remaining parameters we have equations 2.49 to 2.53.

$$< \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) >_{\boldsymbol{\theta},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\alpha}} = \sum_{d,g,k} Z_{dg,k} \quad [< \log \theta_{dk} > \\ -0.5 < \beta_{gk} > (E_{dg}^2 - 2E_{dg} < \mu_{gk} > + < \mu_{gk}^2 >) \\ +0.5 < \log \beta_{gk} >] \tag{2.49}$$

$$\begin{aligned} < \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) >_{\boldsymbol{Z},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\alpha}} &= \sum_{d,k} (< \alpha_k > -1) \log \theta_{dk} + \sum_{d,g,k} < Z_{dg,k} > \log \theta_{dk} \\ &= \sum_{d,k} (< \alpha_k > -1 + \sum_g < Z_{dg,k} >) \log \theta_{dk} \end{aligned} \tag{2.50}$$

$$< \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\mu}) >_{\boldsymbol{\theta},\boldsymbol{Z},\boldsymbol{\beta},\boldsymbol{\alpha}} = -0.5 \sum_{d,g,k} < Z_{dg,k} >< \beta_{gk} > (\mu_{gk}^2 - 2E_{dg}\mu_{gk}) \tag{2.51}$$

$$< \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\beta}) >_{\boldsymbol{\theta},\boldsymbol{Z},\boldsymbol{\mu},\boldsymbol{\alpha}} = \sum_{d,g,k} < Z_{dg,k} > \quad \Big[-0.5\beta_{gk}(< \mu_{gk}^2 > -2E_{dg} < \mu_{gk} > + E_{dg}^2) \\ +0.5 \log \beta_{gk}] \tag{2.52}$$

$$< \log p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\alpha}) >_{\boldsymbol{\theta},\boldsymbol{Z},\boldsymbol{\mu},\boldsymbol{\beta}} = \sum_{d,k} (\alpha_k - 1) < \log \theta_{dk} > \tag{2.53}$$

These expectations take the form of simply the mean $E(X)$), second moments $E(X^2)$ or $E(\log(X))$ and can be evaluated analytically in a standard way. Below we give their values with no working (see [66] for more detail).

- $< \log \theta_{dk} >_{q(\theta)} = \Psi(\tilde{\alpha}_{dk}) - \Psi(\sum_{k'} \tilde{\alpha}_{dk'}) = \log \tilde{\theta}_{dk}$

- $< Z_{dg,k} >_{q(Z)} = r_{dg,k}$
- $< \beta_{gk} >_{q(\beta)} = \tilde{a}_{gk}\tilde{b}_{gk}$
- $< \log \beta_{gk} >_{q(\beta)} = \Psi(\tilde{a}_{gk}) + \log \tilde{b}_{gk}$
- $< \mu_{gk}^2 >_{q(\mu)} = \tilde{m}_{gk}^2 + 1/\tilde{v}_{gk}$
- $< \mu_{gk} >_{q(\mu)} = \tilde{m}_{gk}$
- $< \alpha_k >_{q(\alpha)} = \tilde{t}_{gk}\tilde{s}_{gk}$

For the latent variable $\boldsymbol{Z}$, combining equations 2.49 and 2.45 we have

$$q(\boldsymbol{Z}) = \prod_{dgk} r_{d,k}^{Z_{dg,k}} = Mult(\boldsymbol{Z}; r_{dg,k})$$

$$r_{dg,k} \propto \tilde{\theta}_{d,k} \exp\left[-0.5\tilde{a}_{gk}\tilde{b}_{gk}(E_{dg}^2 - 2E_{dg}\tilde{m}_{gk} + \tilde{m}_{gk}^2 + 1/\tilde{v}_{gk}) + 0.5(\Psi(\tilde{a}_{gk}) + \log \tilde{b}_{gk})\right] \quad (2.54)$$

For the latent variable $\boldsymbol{\theta}$, combining equations 2.50 and 2.44 we have

$$q(\boldsymbol{\theta}) \propto \prod_{dk} \theta_{d,k}^{t_0 s_0 - 1 + \sum_g r_{dg,k}} = Dirichlet(\theta | t_0 s_0 + \sum_g r_{dg,k})$$

For model parameters, recall from equation 2.46 $q(\boldsymbol{\Theta}) \propto \exp(< p(\boldsymbol{E}, \boldsymbol{\theta}, \boldsymbol{Z}|\boldsymbol{\Theta}) >) p(\boldsymbol{\Theta})$. Thus for the posterior distribution of the means $q(\boldsymbol{\mu})$:

$$q(\boldsymbol{\mu}) \propto \prod_{gk} \mathcal{N}(\mu_{gk}; \frac{\sum_d r_{dg,k} E_{dg}}{\sum_d r_{dg,k}}, \tilde{a}_{gk}\tilde{b}_{gk} \sum_d r_{dg,k}) \times \mathcal{N}(\mu_{gk}; m_0, v_0)$$

Combining two Gaussians in the usual way

$$q(\boldsymbol{\mu}) \propto \prod_{gk} \mathcal{N}(\mu_{gk}; \tilde{m}_{gk}, \tilde{v}_{gk})$$

Where

$$\tilde{v}_{gk} = v_0 + \tilde{a}_{gk}\tilde{b}_{gk} \sum_d r_{dg,k}$$

$$\tilde{m}_{gk} = \frac{1}{v_{gk}} \left[v_0 m_0 + \tilde{a}_{gk}\tilde{b}_{gk} \sum_d r_{dg,k} E_{dg} \right]$$

For the posterior distribution of the Dirichlet parameter $q(\boldsymbol{\beta})$

$$p(\boldsymbol{\beta}) \propto \prod_{gk} \beta_{gk}^{a_0 - 1} \exp(-\frac{\beta_{gk}}{b_0})$$

$$q(\boldsymbol{\beta}) \propto \prod_{gk} \Gamma(\boldsymbol{\beta}; 0.5 \sum_d r_{gd,k}, \left[0.5 \sum_d r_{dg,k}((E_{dg} - \tilde{m}_{gk})^2 + 1/\tilde{v}_{gk}) \right]^{-1}) \times \Gamma(\boldsymbol{\beta}; a_0, b_0)$$

$$q(\boldsymbol{\beta}) \sim \prod_{gk} \Gamma(\beta_{gk}; \tilde{a}_{gk}, \tilde{b}_{gk})$$

where

$$\tilde{a}_{gk} = a_0 + 0.5 \sum_d r_{gd,k}$$

$$\frac{1}{\tilde{b}_{gk}} = \frac{1}{b_0} + 0.5 \sum_d r_{dg,k}((E_{dg} - \tilde{m}_{gk})^2 + 1/\tilde{v}_{gk})$$

For the posterior distribution of the precision $q(\boldsymbol{\alpha})$

$$q(\boldsymbol{\alpha}) \sim \prod_k \Gamma(\alpha_k; s_0, \left[\frac{1}{t_0} - \sum_d \log \tilde{\theta}_{dk}\right]^{-1}) = \prod_k \Gamma(\alpha_k; \tilde{s}_k, \tilde{t}_k)$$

The iterative equations of interest for the latent variable parameters are therefore

$$\begin{aligned} \tilde{\alpha}_{dk} &= t_0 s_0 + \textstyle\sum_g r_{dg,k} \\ r_{dg,k} &\propto \tilde{\theta}_{d,k} \exp\left[-0.5\tilde{a}_{gk}\tilde{b}_{gk}(E_{dg}^2 - 2E_{dg}\tilde{m}_{gk} + \tilde{m}_{gk}^2 + 1/\tilde{v}_{gk}) + 0.5(\Psi(\tilde{a}_{gk}) + \log \tilde{b}_{gk})\right] \end{aligned} \tag{2.55}$$

and for the hyper-parameters

$$\begin{aligned} \tilde{v}_{gk} &= v_0 + \tilde{a}_{gk}\tilde{b}_{gk} \textstyle\sum_d r_{dg,k} \\ \tilde{m}_{gk} &= \frac{1}{v_{gk}}\left[v_0 m_0 + \tilde{a}_{gk}\tilde{b}_{gk} \textstyle\sum_d r_{dg,k} E_{dg}\right] \\ \tilde{a}_{gk} &= a_0 + 0.5 \textstyle\sum_d r_{gd,k} \\ \frac{1}{\tilde{b}_{gk}} &= \frac{1}{b_0} + 0.5 \textstyle\sum_d r_{dg,k}((E_{dg} - \tilde{m}_{gk})^2 + 1/\tilde{v}_{gk}) \\ \tilde{s}_k &= s_0 \\ \frac{1}{\tilde{t}_k} &= \frac{1}{t_0} - \textstyle\sum_d \log \tilde{\theta}_{dk} \end{aligned} \tag{2.56}$$

The final form of the update equations given in 2.55 and 2.56 are in-line with what one would expect. The Dirichlet parameter $\tilde{\alpha}_{dk}$ is made up of a *prior mean* count and a number of observations. Similarly the parameters $\tilde{v}_{gk}$, $\tilde{m}_{gk}$, $\tilde{a}_{gk}$, $\tilde{b}_{gk}$, $\tilde{s}_k$ and $\tilde{t}_k$ all decompose into the form

$$\xi_{new} = \xi_{prior} + \xi_{data}$$

for a general parameter ξ.

2.3.6 Evaluation of the Lower Bound

It is useful to be able to evaluate the lower bound on the likelihood $F(\boldsymbol{\Theta})$ given in equation 2.43. Firstly this acts as a test of correct implementation as is should increase with each iteration of the algorithm until convergence. Secondly it can be used as a comparative measure to determine the optimal number of components in a mixture distribution.

$$\begin{aligned} F(\boldsymbol{\Theta}) &= \int q(\boldsymbol{\Theta})q(\boldsymbol{\theta})q(\boldsymbol{Z}) \log \tfrac{p(\boldsymbol{E},\boldsymbol{\theta},\boldsymbol{Z}|\boldsymbol{\Theta})}{q(\boldsymbol{\theta})q(\boldsymbol{Z})} d\boldsymbol{\Theta} d\boldsymbol{\theta} d\boldsymbol{Z} - KL(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta})) \\ &= < \log p(\boldsymbol{E},\boldsymbol{\theta},\boldsymbol{Z}|\boldsymbol{\Theta}) >_{\boldsymbol{\theta},\boldsymbol{Z},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\alpha}} - < \log(q(\boldsymbol{\theta})) >_{\boldsymbol{\theta}} - < \log(q(\boldsymbol{Z})) >_{\boldsymbol{Z}} \\ &\quad - \int q(\boldsymbol{\Theta}) \log \left[\tfrac{q(\boldsymbol{\Theta})}{p(\boldsymbol{\Theta})} \right] dq(\boldsymbol{\Theta}) \end{aligned} \tag{2.57}$$

Evaluating the elements of the bound given in equation 2.57.

$$\begin{aligned} < \log p(\boldsymbol{E},\boldsymbol{\theta},\boldsymbol{Z},|\Theta) >_{\boldsymbol{\theta},\boldsymbol{Z},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\alpha}} &= \textstyle\sum_{d,k}(s_k t_k - 1) < \log \theta_{dk} > \\ &\quad + \textstyle\sum_{d,g,k} r_{dg,k} [< \log \theta_{dk} > \\ &\quad -0.5 a_{gk} b_{gk} (E_{dg}^2 - 2E_{dg} m_{gk} + m_{gk}^2 + 1/v_{gk}) \\ &\quad +0.5(\Psi(a_{gk}) + \log b_{gk})] \end{aligned} \tag{2.58}$$

$$< \log(q(\boldsymbol{\theta})) >_{\boldsymbol{\theta}} = \sum_{dk} (\tilde{\alpha}_{dk} - 1) < \log \theta_{dk} >$$

$$< \log(q(\boldsymbol{Z})) >_{\boldsymbol{Z}} = \sum_{dgk} r_{dg,k} \log r_{dg,k}$$

The $KL(q(\boldsymbol{\Theta})||p(\boldsymbol{\Theta}))$ term decomposes into three terms for the parameter set $\boldsymbol{\Theta} = \{\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\alpha}\}$, these can be analytically evaluated making use of the same identities that were needed in evaluating the expectations earlier. Here, we shall quote the standard results for KL divergences as given in [66].

For the parameter $\boldsymbol{\mu}$, $p(\boldsymbol{\mu}) \sim \prod_{gk} \mathcal{N}(\mu_{gk}; m_0, v_0)$ and $q(\boldsymbol{\mu}) \sim \prod_{gk} \mathcal{N}(\tilde{m}_{gk}, \tilde{v}_{gk})$.

$$\begin{aligned} KL(q(\boldsymbol{\mu})||p(\boldsymbol{\mu})) &= \textstyle\sum_{gk} 0.5 \log \frac{v_{gk}}{v_0} + 0.5 v_0 \left[m_{gk}^2 + m_0^2 + 1/v_{gk} - 2 m_{gk} m_0 \right] - 0.5 \\ &= \textstyle\sum_{gk} 0.5 \log \frac{v_{gk}}{v_0} + 0.5 v_0 \left[m_{gk} - m_0 \right]^2 + 0.5 \left[\frac{v_0}{v_{gk}} - 1 \right] \end{aligned} \tag{2.59}$$

Where we have grouped the corresponding terms to show $KL = 0$ when the parameters from the two distributions are equal. For the parameter $\boldsymbol{\beta}$, $p(\boldsymbol{\beta}) \sim \prod_{gk} \Gamma(\beta_{gk}; a_0, b_0)$ and $q(\boldsymbol{\beta}) \sim \prod_{gk} \Gamma(\tilde{a}_{gk}, \tilde{b}_{gk})$

$$\begin{aligned} KL(q(\boldsymbol{\beta})||p(\boldsymbol{\beta})) = \textstyle\sum_{gk} \quad & (\tilde{a}_{gk} - 1)\Psi(\tilde{a}_{gk}) - \log \tilde{b}_{gk} - \tilde{a}_{gk} - \log \Gamma(\tilde{a}_{gk}) \\ & + \log \Gamma(a_0) + a_0 \log b_0 - (a_0 - 1)(\Psi(\tilde{a}_{gk}) + \log \tilde{b}_{gk}) + \tfrac{\tilde{a}_{gk} \tilde{b}_{gk}}{b_0} \end{aligned} \tag{2.60}$$

Again we shall group the terms to show that $KL = 0$ when the parameter from the two distributions are equal.

$$\begin{aligned} KL(q(\boldsymbol{\beta})||p(\boldsymbol{\beta})) = \textstyle\sum_{gk} \quad & (\tilde{a}_{gk} - a_0)\Psi(\tilde{a}_{gk}) + a_0(\log b_0 - \log \tilde{b}_{gk}) \\ & + \log \Gamma(a_0) - \log \Gamma(\tilde{a}_{gk}) + \tilde{a}_{gk} \left[\tfrac{\tilde{b}_{gk}}{b_0} - 1 \right] \end{aligned} \tag{2.61}$$

For the parameter $\boldsymbol{\alpha}$, $p(\boldsymbol{\alpha}) = \prod_k p(\alpha_k) \sim \prod_k \Gamma(\alpha_k; t_0, s_0)$ and $q(\boldsymbol{\alpha}) = \prod_k q(\alpha_k) \sim \prod_k \Gamma(t_k, s_k)$

$$\begin{aligned} KL(q(\boldsymbol{\alpha})||p(\boldsymbol{\alpha})) = \textstyle\sum_{k} \quad & (\tilde{t}_k - 1)\Psi(\tilde{t}_k) - \log \tilde{s}_k - \tilde{t}_k - \log \Gamma(\tilde{t}_k) \\ & + \log \Gamma(t_0) + t_0 \log s_0 - (t_0 - 1)(\Psi(\tilde{t}_k) + \log \tilde{s}_k) + \tfrac{\tilde{t}_k \tilde{s}_k}{s_0} \end{aligned} \tag{2.62}$$

Again grouping the terms, and noting that $s_k = s_0$

$$KL(q(\boldsymbol{\alpha})||p(\boldsymbol{\alpha})) = \sum_k \begin{array}{l} (\tilde{t}_k - 1)\Psi(\tilde{t}_k) - \log \tilde{s}_k - \tilde{t}_k - \log \Gamma(\tilde{t}_k) \\ + \log \Gamma(t_0) + t_0 \log s_0 - (t_0 - 1)(\Psi(\tilde{t}_k) + \log \tilde{s}_k) + \frac{\tilde{t}_k \tilde{s}_k}{s_0} \end{array} \tag{2.63}$$

We shall not give an application of the Variational Bayes approach to parameter estimation in LPD here but shall use the results here in an application to Microarray data given in chapter 6.

2.3.7 Monte Carlo Methods

Excellent introductions to Monte Carlo methods are given in [56] and [6]. Here we shall give a brief overview of Monte Carlo Methods, in particular the Metropolis Method and the Gibbs Sampler. Monte Carlo Methods are now a commonly used tool for performing inference in probabilistic models. This class of methods was first introduced in [60] and later extended in [89]. The methods and their variations are used to solve two main problems in inference: Sampling and Integration. In sampling we wish to generate samples

$$\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$

from a density $P(\mathbf{x})$. In integration we want to construct a good approximation to the expectation of a function $f(\mathbf{x})$ over a density $P(\mathbf{x})$:

$$< f(\mathbf{x}) >= \int d\mathbf{x} P(\mathbf{x}) f(\mathbf{x})$$

It is commonplace in inference to want to sample from the posterior distribution. In many cases this is not in a convenient form that allows easy sampling. This is often due to the complexity of a normalising factor. For example, given the un-normalised density

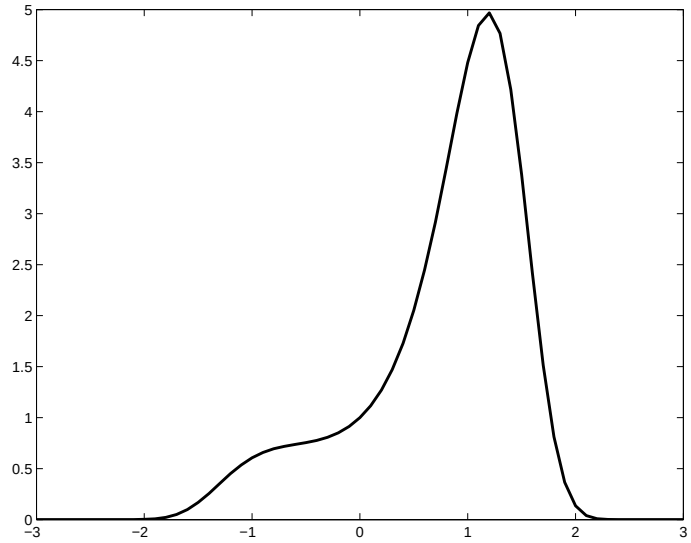


Fig. 2.15: Density $\tilde{P}(x)$

$$\tilde{P}(x) = \exp(x^2 - x^4/2 + x)$$

It is straightforward to evaluate $\tilde{P}(x)$ and hence plot (see figure 2.15)

But to generate samples from the true density

$$P(x) = \frac{\tilde{P}(x)}{Z} \tag{2.64}$$

$$Z = \int \exp(x^2 - x^4/2 + x)dx \tag{2.65}$$

the normalising factor Z in equation 2.65 must be calculated. This is often hard.

For the problem of integration, a naive approach would be to explore some or all of the state space uniformly and average over these states.

$$< f(\mathbf{x}) > \simeq \frac{1}{N} \sum_{i}^{N} f(x_i)$$

Even for a simple discrete density this often requires an unfeasibly large number of calculations. For the example given in figure 2.15 the range of the density is $[-\infty, \infty]$ but the range that has any significance is arguably $[-3, 3]$. Under a naive approach computational time would be wasted on insignificant areas of the state space. The idea of Monte Carlo methods is to explore the state space in a more orderly way. Variations on this idea include *Rejection Sampling*, *Importance Sampling* and the famous *Metropolis Method* [56]. We shall now concentrate on the *Metropolis Method*.

Markov Chain Monte Carlo

A Markov chain is a discrete time stochastic process. It has the property that its sequence of random variables are conditionally independent up until the previous point.

$$P(x_N|x_1, x_2, \ldots, x_{N-1}) = P(x_N|x_{N-1}) \tag{2.66}$$

The conditional distribution $P(x_N|x_{N-1})$ is called the *transition probability* of the process and governs the evolution of the Markov chain. In the usual way can integrate over the conditioned variable to obtain the marginal distribution.

$$P(x_N) = \int P(x_N|x_{N-1})P(x_{N-1})dx_{N-1} \tag{2.67}$$

For a given Markov Chain there may exist a stationary, or equilibrium, distribution which has the property

$$\pi(X) = \int P(X|Y)\pi(Y)dY \tag{2.68}$$

The *Metropolis* method is a Markov Chain Monte Carlo method. It constructs a Markov Chain that has an equilibrium distribution, given by equation 2.68, equal to our target distribution $P(x)$.

First we assume that we can easily calculate the un-normalised density $\tilde{P}(x)$ given in equation 2.64. We then decide on a proposal density $q(y; x^{(t)})$. This density dictates where we look to make the transition $x^{(t)} \rightarrow y$, and is dependent on the current state $x^{(t)}$. This density q in theory can be anything. But a simple choice could be a Gamma distribution, mean centred on $x^{(t)}$. We then pick an initial state x_0 at random. The *Metropolis* Method then proceeds as follows:

Sample

$$y \sim q(y; x^{(t)})$$

Compute

$$a = \frac{\tilde{P}(y)q(x^{(t)}; y)}{\tilde{P}(x^{(t)})q(y; x^{(t)})}$$

- if $a \geq 1$ then $x^{(t+1)} = y$
- else $x^{(t+1)} = y$ with probability a and $x^{(t+1)} = x^{(t)}$ with probability $1 - a$

As can be seen from above the normalising constant Z from equation 2.64 does not come into the method. It has been effectively cancelled as we are only considering ratios of states. It can be shown that for a choice of $q(x; y) > 0$ and as $t \rightarrow \infty$ the distribution of $x^{(t)} \rightarrow P(x)$. Due to the Markov nature of the samples $x^{(t)}$, successive samples will be correlated. This will mean a large number of iterations are needed for the distribution to have converged sufficiently well.

The Gibbs Sampler

The *Gibbs Sampler* can be seen as a special case of the *Metropolis* method. It is the case when the proposal density $q(x; y)$, for the random variable y, is the full conditional distribution for y. Additionally each new state is always accepted with probability 1. More thoroughly, suppose

$$\mathbf{x} = (x_1, \ldots, x_n)$$

is an N dimensional vector of parameters. The full conditional of x_k will be

$$P(x_k|x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_n) = \frac{\pi(x_1, \ldots, x_{k-1}, x_k, x_{k+1}, \ldots, x_n)}{\int \pi(x_1, \ldots, x_{k-1}, x_k, x_{k+1}, \ldots, x_n) dx_k} \tag{2.69}$$

For a full Gibbs sampler the full conditional is needed for each element of $\mathbf{x}$. The algorithm is repeated as follows, sample:

$$\begin{aligned} Y_1 &\sim \pi(x_1|x_2, x_3, \ldots, x_n) \\ Y_2 &\sim \pi(x_2|x_1, x_3, \ldots, x_n) \\ &\vdots \\ Y_n &\sim \pi(x_n|x_1, x_2, \ldots, x_{n-1}) \end{aligned} \tag{2.70}$$

Update:

$$\begin{aligned} x_1 &= Y_1 \\ x_2 &= Y_2 \\ &\vdots \\ x_n &= Y_n \end{aligned} \tag{2.71}$$

2.4 Examples

We shall now give three examples on how to perform inference. The first is an example of the *Expectation Maximisation* algorithm applied to the Gaussian Mixture model given in the graphical model in figure 2.5. The second is a *Gibbs Sampler* for the same model given in figure 2.5, and finally the third is a hybrid *Gibbs Sampler - Metropolis Hastings* approach to the Latent Process Decomposition model ([70]) given in the graphical model in figure 2.6.

2.4.1 Example 1: EM for a Gaussian Mixture Model

We start with a set of data samples $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_d\}$. In a Gaussian Mixture Model, each I dimensional sample $\mathbf{x}_i$ is assumed to have been drawn from one of k distinct classes. Each class is assigned a set of distinct mean μ_k and covariance $\mathbf{\Sigma}_k$ parameters. Additionally there is an overall mixing parameter π_k which indicates the total proportion of samples coming from each class. The total set of model parameters is then:

$$\theta = \{\mu, \mathbf{\Sigma}, \pi\}$$

The origin, that is the class from which it was derived, of each sample is unknown. We therefore additionally define the set of parameters $Z = \{Z_1, \ldots, Z_d\}$ that indicate from which class a sample $\mathbf{x}_i$ was generated. In particular, $Z_i = k$ indicates that sample i came from class k. The first step is to derive an expression for the *Completed Data* likelihood $p(Z, X|\theta)$, originally given in equation 2.30.

$$p(Z, X|\theta) = \prod_d p(Z_d, \mathbf{x}_d|\theta) \tag{2.72}$$

As each sample has come from once specific class, $\delta_{dk} = \delta(Z_d, k)$ will have a single non-zero element at the point when $Z_d = k$, so $\delta_{dk} = P(Z_{dk} = 1)$. Thus,

$$\begin{aligned} p(Z_d, x_d|\theta) &= \textstyle\sum_k \delta_{dk} p(Z_d = k, x_d|\theta) \\ &= \textstyle\sum_k \delta_{dk} p(\delta_{dk} = 1, x_d|\theta) \\ &= \textstyle\sum_k \delta_{dk} p(\delta_{dk} = 1|\theta) p(x_d|\delta_{dk} = 1, \theta) \\ &= \textstyle\sum_k \delta_{dk} p(\delta_{dk} = 1|\pi) p(x_d|\delta_{dk} = 1, \mu_k, \mathbf{\Sigma} =_k) \\ &= \textstyle\sum_k \delta_{dk} \pi_k p(x_d|Z_d = k, \mu_k, \mathbf{\Sigma}_k) \end{aligned} \tag{2.73}$$

As there is only one non-zero element of δ_{dk}, and it is equal to unity, we can log equation 2.73 and bring the summation to the front, giving:

$$\log(Z_d, x_d|\theta) = \sum_k \delta_{dk} \log\left[\pi_k p(x_d|Z_d = k, \mu_k, \mathbf{\Sigma}_k)\right] \tag{2.74}$$

So in total, combining equations 2.72 and 2.74.

$$\begin{aligned} \log(Z, X|\theta) &= \textstyle\sum_{d,k} \delta_{dk} \log\left[\pi_k p(x_d|Z_d = k, \mu_k, \mathbf{\Sigma}_k)\right] \\ &= \textstyle\sum_{d,k} P(Z_{dk} = 1) \log\left[\pi_k p(x_d|Z_d = k, \mu_k, \mathbf{\Sigma}_k)\right] \end{aligned} \tag{2.75}$$

We can now evaluate $Q(\theta|\theta_n)$ given in equation 2.33.

$$\begin{aligned} Q(\theta|\theta_n) &= E_Z\left[\log p(Z, X|\theta)|X, \theta_n\right] \\ &= E_Z\left[P(Z_{dk} = 1) \log\left(\pi_k p(x_d|Z_d = k, \mu_k, \mathbf{\Sigma}_k)\right)|X, \theta_n\right] \\ &= \textstyle\sum_{d,k} E_Z\left[P(Z_{dk} = 1|x_d, (\theta_n)_k)\right] \log\left(\pi_k p(x_d|Z_d = k, \mu_k, \mathbf{\Sigma}_k)\right) \end{aligned} \tag{2.76}$$

Take $(\theta_n)_k = \theta_{nk}$ to be the set of parameters for class k after n iterations, so $(\pi_n)_k = \pi_{nk}$ and likewise for μ and Σ. We can re-express the expectation via Bayes theorem:

$$E_Z\left[P(Z_{dk} = 1|x_d, \theta_{nk})\right] = P(Z_{dk} = 1|x_d, \theta_n) \tag{2.77}$$

$$\frac{P(x_d|Z_{dk}=1,\theta_{nk})P(Z_{dk})}{P(x_d)} = \frac{P(x_d|Z_{dk}=1,\mu_{nk},\Sigma_{nk})\pi_{nk}}{\sum_{k'} P(x_d|Z_{dk'}=1,\theta_{nk})\pi_{nk'}} \tag{2.78}$$

For convenience we shall write

$$\gamma_{nk}(\mathbf{x}_d) = P(\mathbf{x}_d|Z_{dk}=1,\mu_{nk},\Sigma_{nk})\pi_{nk} \tag{2.79}$$

combining equations 2.76, 2.77 and 2.78.

$$Q(\theta|\theta_n) = \sum_{d,k} \frac{\gamma_{nk}(\mathbf{x}_d)\log\left(\pi_k p(\mathbf{x}_d|Z_d=k,\mu_k,\mathbf{\Sigma}_k)\right)}{\sum_{k'}\gamma_{nk'}} \tag{2.80}$$

For simplicity we shall assume that Σ is isotropic. That is, it only has diagonal elements. The logged density can then be written:

$$\log p(x_d|\mu_k,\mathbf{\Sigma}_k) = \sum_i \frac{-(x_{di}-\mu_{ki})^2}{2\sigma_{ki}^2} - \log(\sigma_{ki}) - \frac{\log(2\pi)}{2} \tag{2.81}$$

Using equations 2.80 and 2.81 we can now maximise $Q(\theta|\theta_n)$ for the parameters μ, σ and π

$$\frac{\partial Q(\theta|\theta_n)}{\partial \mu_{ki}} \propto \sum_d \gamma_{nk}(\mathbf{x}_d)(x_{di}-\mu_{ki}) \tag{2.82}$$

$$\frac{\partial Q(\theta|\theta_n)}{\partial \sigma_{ki}} \propto \frac{\sum_d (x_{di}-\mu_{ki})^2}{\sigma_{ki}^3} - \frac{1}{\sigma_{ki}} \tag{2.83}$$

π is a multinomial parameter and so is constrained by $\sum_k \pi_k = 1$, the maximisation is therefore restricted by using a Lagrange multiplier λ

$$\frac{\partial}{\partial \pi_k}\left[Q(\theta|\theta_n) - \lambda(\sum_k \pi_k - 1)\right] \propto \frac{\sum_d \gamma_{nk}(\mathbf{x}_d)}{\pi_k} - \lambda \tag{2.84}$$

Zeroing these gradients we are left with the equations:

$$\mu_{ki} = \frac{\sum_d \gamma_{nk}(\mathbf{x}_d)(x_{di})}{\sum_d \gamma_{nk}(\mathbf{x}_d)} \tag{2.85}$$

$$\sigma_{ki} = \frac{\sum_d \gamma_{nk}(\mathbf{x}_d)(x_{di} - \mu_{ki})^2}{\sum_d \gamma_{nk}(\mathbf{x}_d)} \tag{2.86}$$

$$\pi_k \propto \sum_d \gamma_{nk}(\mathbf{x}_d) \tag{2.87}$$

The EM algorithm for a Gaussian Mixture Model is thus:

- E-Step: Calculate $\gamma_{nk}(\mathbf{x}_d)$ in equation 2.79.
- M-Step: Update μ, σ and π using equations 2.85, 2.86 and 2.87.

We shall now give a very simple practical example of the EM algorithm applied to one dimensional data. Figure 2.16 shows a histogram of artificially generated data. This data was sampled from 3 distinct Gaussian distributions with Means 0, 5 and 10. The EM algorithm was then run to fit a number of Gaussian distributions to the data. Figure 2.17 shows the results of a 3 component mixture model. The continuous lines show the densities of the estimated distributions. As the data was originally generated by a 3 component mixture the estimated distributions provide a good fit to the data. Figure 2.18 shows the same results but for a 5 component mixture, additionally the combined distribution is plotted. With an additional two components the mixture model will more closely fit the data, this is however not necessarily desirable as there is some clear over-fitting.

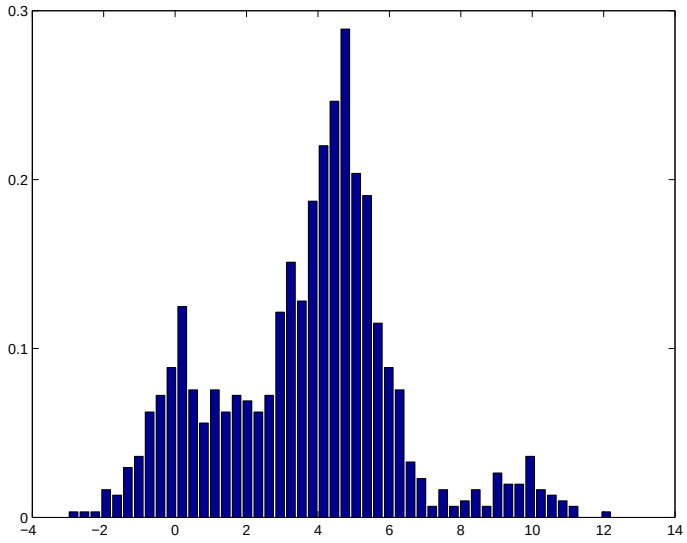


Fig. 2.16: Artificially Generated Data from 3 Gaussians

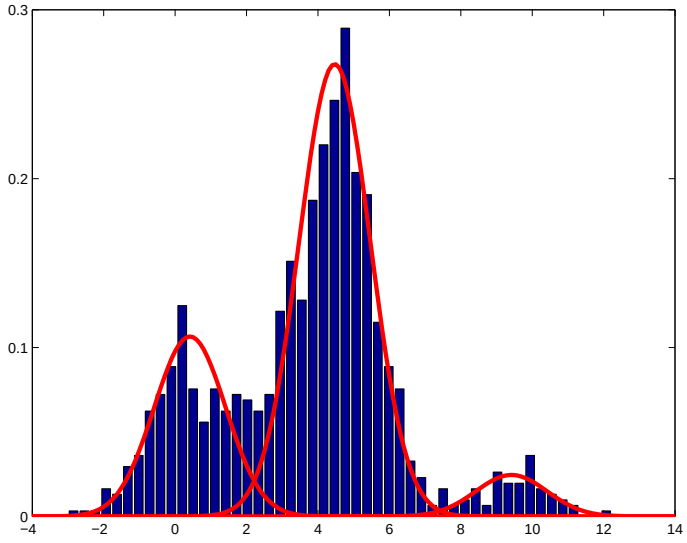


Fig. 2.17: A 3 Component Mixture Derived using the EM algorithm

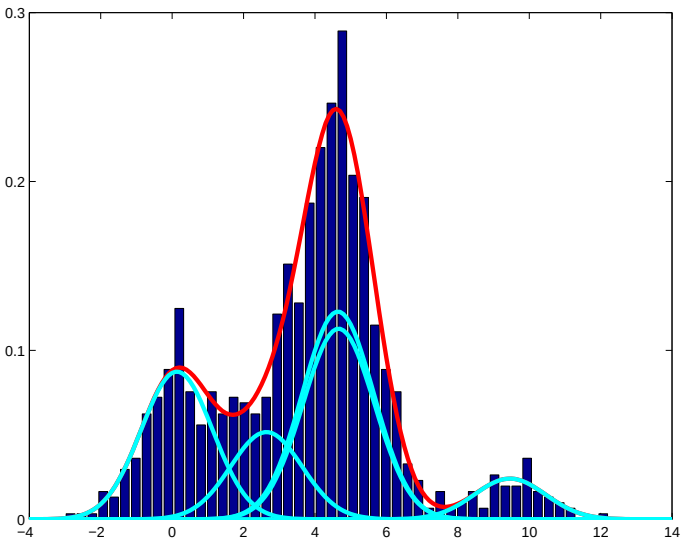


Fig. 2.18: A 5 Component Mixture Derived using the EM algorithm. The continuous plot gives the combined mixture density.

2.4.2 Example 2: Mixture Model Gibbs Sampler

We shall now derive a Gibbs sampler for the model given in figure 2.5. The approach given here is very standard. As described in section 2.3.7 the Gibbs sampler requires sampling from the full conditional distribution for each parameter. To ensure ease of sampling conjugate prior distributions have been chosen.

As before the Mixture model is defined by a set of model parameters.

$$\Theta = (\pi, \mu, \sigma)$$

The likelihood of the data given the model parameters is:

$$L(\mathbf{x}|\Theta) = \prod_d \sum_k \pi_k P(x_d|\mu_k, \sigma_k^2)$$

π is a discrete mixing parameter and has a uniform Dirichlet prior.

$$P(\pi) \sim \frac{\Gamma(K \times \alpha)}{\Gamma(\alpha)^K} \prod_k^K \pi_k^{\alpha-1}$$

μ is the mean of a Gaussian distribution as has a zero mean Gaussian prior.

$$P(\mu) \sim N(0, \tau^2)$$

σ^2 is the variance of a Gaussian an has an Inverse Gamma prior.

$$P(\sigma^2) \sim \frac{s_2^{s_1}}{\Gamma(s_1)\sigma^{2(s_1+1)}} \exp\left(\frac{-s_2}{\sigma^2}\right)$$

The joint Posterior of the parameters can be written

$$\begin{aligned} P(\pi, \mu, \sigma|\mathbf{x}, \tau, s, \alpha) &= P(\pi, \mu, \sigma, \mathbf{x}|\tau, s, \alpha)/P(\mathbf{x}) \\ &\propto P(\pi, \mu, \sigma, \mathbf{x}|\tau, s, \alpha) \\ &= P(x|\pi, \mu, \sigma)P(\pi|\alpha)P(\mu|\tau)P(\sigma|s) \end{aligned} \tag{2.88}$$

$$\begin{aligned} P(\pi, \mu, \sigma|\mathbf{x}, \alpha, \tau, s) \propto \; & \prod_d \left[\sum_k \pi_k \tfrac{1}{\sigma_k} \exp(-\tfrac{1}{2\sigma_k^2}(x_d - \mu_k)^2)\right] \\ & \times \prod_k \pi_k^{\alpha-1} \prod_k \exp(-\tfrac{\mu_k^2}{2\tau^2}) \prod_k \frac{\exp\left(\frac{-s_2}{\sigma_k^2}\right)}{\sigma_k^{2(s_1+1)}} \end{aligned} \tag{2.89}$$

A common technique for simplifying posteriors for Mixture distributions is to introduce model indicators. γ_d indicates which mixture sample x_d came from.

$$\gamma_d \in (1, 2, ..., k)$$

The likelihood is now written:

$$L(\mathbf{x}|\Theta) = \prod_d P(x_d|\mu_{\gamma_d}, \sigma^2_{\gamma_d})$$

and

$$P(\gamma = k) = \pi_k$$

$$P(\gamma_1, ...\gamma_d|\pi_1, ..., \pi_k) = \prod_d P(\gamma_d|\pi) = \prod_k \pi_k^{n_k}$$

with n_k number of samples coming from mixture k. The posterior from equation 2.88 is now written

$$\begin{aligned} P(\gamma, \mu, \sigma^2, x) &\propto P(x|\pi, \mu, \sigma)P(\gamma|\pi)P(\mu)P(\sigma)P(\gamma)P(\pi) \\ &\propto \textstyle\prod_d \left[\frac{1}{\sigma_{\gamma_d}} \exp(-\frac{1}{2\sigma^2_{\gamma_d}}(x_d - \mu_{\gamma_d})^2)\right] \\ &\times \textstyle\prod_k \pi_k^{n_k} \prod_k \exp(-\frac{\mu_k^2}{2\tau^2}) \prod_k \frac{\exp\left(\frac{-s_2}{\sigma_k^2}\right)}{\sigma_k^{2(s_1+1)}} \prod_k \pi_k^{\alpha-1} \end{aligned} \tag{2.90}$$

The conditional distribution for each parameter is proportional to the joint distribution. By reading off each conditional from equation 2.90 and rearranging we have

$$P(\pi|\gamma, \mu, \sigma^2, \mathbf{x}) \propto \prod_k \pi_k^{n_k} \prod_k \pi_k^{\alpha-1}$$

$$P(\pi|\gamma, \mu, \sigma^2, \mathbf{x}) \sim Dirichlet(n_1 + \alpha - 1, \ldots, n_k + \alpha - 1) \tag{2.91}$$

$$P(\mu|\gamma, \pi, \sigma^2, \mathbf{x}) \propto \prod_d \left[\frac{1}{\sigma_{\gamma_d}} \exp(-\frac{1}{2\sigma^2_{\gamma_d}}(x_d - \mu_{\gamma_d})^2)\right] \prod_k \exp(-\frac{\mu_k^2}{2\tau^2})$$

$$P(\mu_k|\gamma,\pi,\sigma^2,\mathbf{x}) \sim \mathcal{N}\left(\frac{\tau^2\sum_d \delta(\gamma_d,k)x_d}{n_k\tau^2+\sigma^2}, \left(\frac{n_k}{\sigma^2}+\frac{1}{\tau^2}\right)^{-1}\right) \tag{2.92}$$

$$P(\sigma^2|\gamma,\pi,\mu,\mathbf{x}) \propto \prod_d \left[\frac{1}{\sigma_{\gamma_d}}\exp(-\frac{1}{2\sigma^2_{\gamma_d}}(x_d-\mu_{\gamma_d})^2)\right]\prod_k \frac{\exp\left(\frac{-s_2}{\sigma_k^2}\right)}{\sigma_k^{2(s_1+1)}}$$

$$P(\sigma_k^2|\gamma,\pi,\mu,\mathbf{x}) \sim InverseGamma\left(s_2+\frac{D}{2}, s_1+\frac{1}{2}\sum_d \delta(\gamma_d,k)(x_d-\mu_{\gamma_d})^2\right) \tag{2.93}$$

Using $\prod_k \pi_k^{n_k} = \prod_d \pi_{\gamma_d}$

$$P(\gamma|\pi,\mu,\sigma^2,\mathbf{x}) \propto \prod_d \pi_{\gamma_d}\left[\frac{1}{\sigma_{\gamma_d}}\exp(-\frac{1}{2\sigma^2_{\gamma_d}}(x_d-\mu_{\gamma_d})^2)\right]$$

$$P(\gamma_d=k|\pi,\mu,\sigma^2,\mathbf{x}) \propto Mult\left(\pi_k\exp(-\frac{1}{2\sigma_k^2}(x_d-\mu_k)^2)\right) \tag{2.94}$$

Notice how through the use of conjugate priors each *Posterior* is a member of the same class of distributions as its corresponding *Prior*. We now give a simple demonstration of a Gibbs Sampler. 100 one dimensional random variables were sampled from a Mixture model with the following parameters $\mu_1 = 1$, $\mu_2 = 5$ and $\mu_1 = 10$ $\sigma_1 = \sigma_2 = \sigma_3 = 1$, $\pi_1 = 0.25, \pi_2 = 0.65, \pi_3 = 0.1$. The Gibbs sampler given by equations 2.91 to 2.94 was then run for 2000 iterations with a *burn in* of 1000 iterations. The burn in period is a sequence of N iterations that discarded and do not contribute to the posterior distributions. There is some debate as to whether or not a burn in period is strictly necessary (see [1]), but at the least it provides a suitable initialisation for the parameters in the model. The next 1000 iterations form the samples which will make up the posterior distributions. The choice of 1000 is somewhat arbitrary but as a obvious general rule the minimum number of iterations increases with the number of parameters (ie number of posterior distributions) in the model.

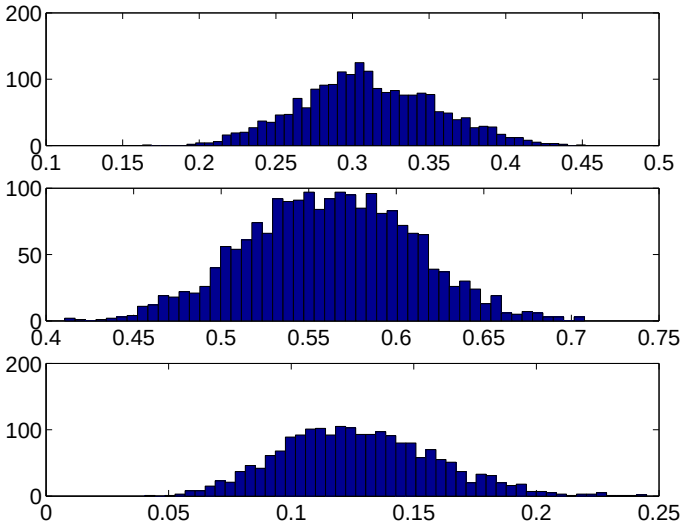


Fig. 2.19: Histogram of the posterior distribution for π

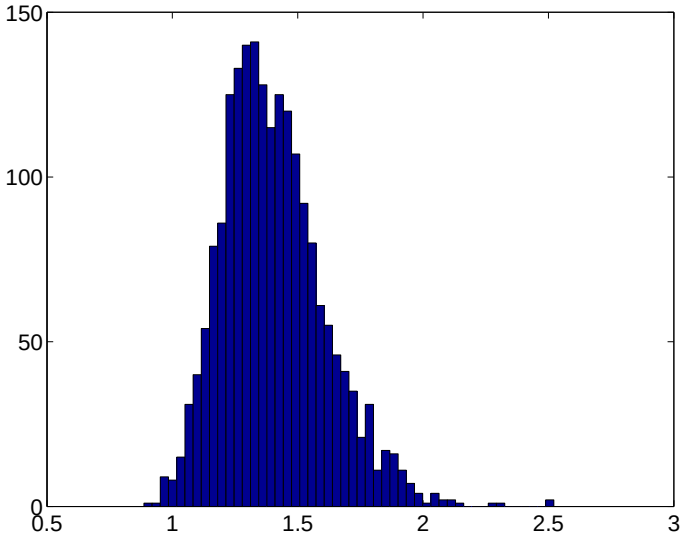


Fig. 2.20: Histogram of the posterior distribution for σ^2

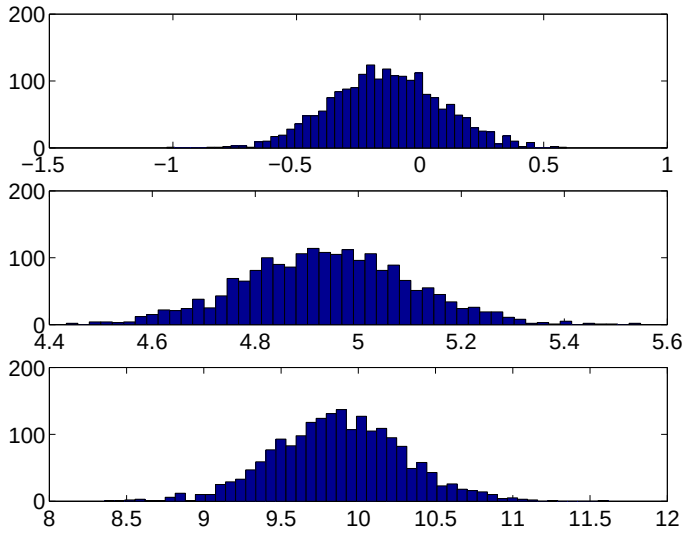


Fig. 2.21: Histogram of the posterior distribution for each μ

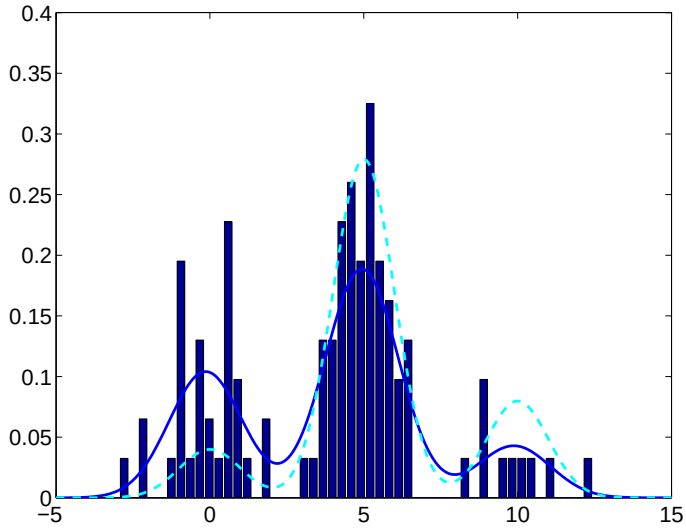


Fig. 2.22: Normalised histogram of original data also showing the inferred density as a bold line and the actual density as a dashed line

Figures 2.19 to 2.21 show histograms represents the posterior density of the model parameters π, σ^2 and μ. Figure 2.22 shows a histogram of the original data. Additionally plotted, in a bold line, is a mixture density using the means of the posterior of μ, σ and π and in a dashed line a mixture density for the original parameters. As can be seen the posterior densities are all peaked near the values of the original parameters used to generate the data. By comparison to the results from the EM algorithm in section 2.4.1 we see how much more information the Gibbs sampler gives us by providing the full posterior distributions rather than just a point estimates.

2.4.3 Example 3: LPD Gibbs Sampler

We have introduced the graphical model for the Latent Process Decomposition ([70]) algorithm in section 2.2. A Markov chain Monte Carlo algorithm for performing partial inference, that is with some model parameters remaining fixed, for the related LDA model is given in [39]. We shall give the full derivation for LPD here. As in the previous example to construct a Gibbs Sampler we need to derive full conditional distributions for each of the parameters.

The following distributions will be needed.

$$\begin{aligned}
&P(\alpha|\mu,\sigma,\theta,Z,E)\\
&P(\theta_d|\alpha,\mu,\sigma,\theta_{-d},Z,E)\\
&P(Z_{dg}|\alpha,\mu,\sigma,Z_{-(dg)},\theta,E)\\
&P(\mu_{gk}|\alpha,\mu_{-(gk)},\sigma,\theta,Z,E)\\
&P(\sigma_{gk}|\alpha,\mu,\sigma_{-(gk)},\theta,Z,E)
\end{aligned} \tag{2.95}$$

Each of these conditional distributions is proportional to the joint, which can be factored as follows:

$$P(Z,\theta,E,\mu,\sigma,\alpha) = P(Z,\theta,E|\mu,\sigma,\alpha)P(\mu)P(\sigma)P(\alpha)$$

$$P(Z, \theta, E|\mu, \sigma, \alpha) = P(\theta|\alpha) \prod_g P(Z_g|\theta) P(E_g|Z_g, \mu, \sigma)$$

In full

$$P(Z, \theta, E, \mu, \sigma, \alpha) = \prod_d \frac{\Gamma(\sum_h \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_{k'} \theta_{dk'}^{\alpha_{k'}-1} \prod_{k,g} \left(\theta_{dk} \frac{1}{\sqrt{2\pi\sigma_{gk}^2}} \exp(-\frac{(E_{dg} - \mu_{gk})^2}{2\sigma_{gk}^2}) \right)^{I_{\{Z_{dg}=k\}}} \tag{2.96}$$

Where $I_{\{Z_{dg}=k\}}$ is an indicator variable giving the process chosen for the g'th element of example d. Conjugate prior distributions are taken to allow easy sampling from the posterior.

$$P(\alpha) \sim Gamma(a, b) = \frac{1}{\Gamma(a) b^a} \alpha^{a-1} e^{-x/\alpha}$$

$$P(\mu_{gk}) \sim \mathcal{N}(0, \tau) = \frac{1}{\sqrt{2\pi\tau^2}} \exp(-\frac{\mu_{gk}^2}{2\tau^2})$$

$$P(\sigma_{gk}^2) \sim InverseGamma(s_1, s_2) = \frac{s_2^{s_1}}{\Gamma(s_1)\sigma^{2(s_1+1)}} \exp\left(\frac{-s_2}{\sigma^2}\right)$$

Combining these prior distributions with the joint given in equation 2.96 we have the following distributions for those given in equation 2.95.

$$\begin{aligned} P(\alpha|\theta) &\propto P(\theta|\alpha) P(\alpha) \\ &\propto \textstyle\prod_k e^{(\alpha_k - 1)\log\theta_{dk}} \frac{1}{\Gamma(a) b^a} \alpha_k^{a-1} e^{-\alpha_k/b} \\ &\sim Gamma(a, \hat{b}) \end{aligned} \tag{2.97}$$

Where

$$\hat{b} = \left(\frac{1}{b} - \sum_d \log(\theta_{dk}) \right)^{-1}$$

$$\begin{aligned} P(\theta|\alpha, Z) &\propto \prod_{dk} \theta_{dk}^{\alpha_k + \sum_g I_{\{Z_{dg}=k\}}} \\ P(\theta_d|\alpha, Z) &\sim Dirichlet(\alpha_1 + \sum_g I_{\{Z_{dg}=1\}}, \ldots, \alpha_k + \sum_g I_{\{Z_{dg}=k\}}) \end{aligned} \tag{2.98}$$

$$\begin{aligned} P(Z_{dg}|\alpha, \mu, \sigma, Z_{-(dg)}, \theta, E) &= P(Z_{dg}|\mu, \sigma, \theta_d, E_{dg}) \\ &\propto \prod_k \left(\theta_{dk} \frac{1}{\sqrt{2\pi\sigma_{gk}^2}} \exp(-\frac{(E_{dg}-\mu_{gk})^2}{2\sigma_{gk}^2}) \right)^{I_{\{Z_{dg}=k\}}} \end{aligned} \tag{2.99}$$

Hence, Z_{dg} is a multinomial, $M(\gamma, 1)$ with $\gamma_k = \theta_{dk} \frac{1}{\sqrt{2\pi\sigma_{gk}^2}} \exp(-\frac{(E_{dg}-\mu_{gk})^2}{2\sigma_{gk}^2})$.

Let t index the d st $Z_{dg} = k$ with $T = \sum_d I_{\{Z_{dg}=k\}}$

$$\begin{aligned} P(\mu_{gk}|\alpha, \mu_{-(gk)}, \sigma, \theta, Z, E) &= P(\mu_{gk}|\sigma_{gk}, \theta_d, Z_{.g}, E_{.g}) \\ &\sim \mathcal{N}(\frac{\tau^2 \sum_t E_{tg}}{T\tau^2 + \sigma_{gk}^2}, (\frac{T}{\sigma_{gk}^2} + \frac{1}{\tau^2})^{-1}) \end{aligned} \tag{2.100}$$

$$P(\sigma_{gk}^2|\alpha, \mu, \sigma_{-(gk)}, \theta, Z, E) \sim InverseGamma\left(s_2 + \frac{T}{2}, s_1 + \frac{1}{2} \sum_t (E_{tg} - \mu_{gk})^2 \right) \tag{2.101}$$

The full derivation for the conditional distributions of μ and σ is given in Appendix A. We shall not give an application of this Gibbs Sampler here but shall use the results here in an application to Microarray data given in chapter 6.

Chapter 3

Deriving a Hierarchical Representation of Lung Disease using Re-Sampling Mixture Models

3.1 Introduction

A common task in medical image processing is to segment images. That is to separate parts of the image into separate regions defined by appearance, function, the presence of disease or some other criteria. Viewing individual medical images as made up of a number of small cells, typically 4×4 pixels or larger, one view of segmentation is of clustering these small cells into self similar groupings. In this section we present a novel extension to the Latent Dirichlet Allocation (LDA) algorithm ([15]). This model is a generative probabilistic model which clusters Computed Tomography (CT) chest scans to give a hierarchical representation of disease.

This research in this chapter was carried out as an initial investigation into using decomposable probabilistic models on image data, with this in mind there is equal emphasis put on

the mathematical techniques and the application. This work was presented at and appeared in the proceedings of MIUA - *Medical Image Understanding and Analysis 2005 - University of Bristol*, 19-20 July, 2005, [21].

3.2 Motivation

In recent years there has been an increased interest in the classification of medical images [76], [82]. This has been due to increased availability of data and also increased processing power that enables real time computer aided diagnosis. Most of this work has been in the area of supervised classification. That is, learning to classify distinct tissue types based on labelled examples. This is undoubtedly a sensible way to proceed. However it requires a substantial quantity of hand labelled data which was unfortunately not available. It also, in a sense only confirms what we already know. For these reasons we will proceed in an entirely unsupervised manner. We describe some intuitive qualities of LDA which confirm its suitability as a technique for learning from Radiological data. LDA is a state of the art generative hierarchical Bayesian model which can be thought of as a generalisation of traditional mixture models. The mixing quantities, rather than being fixed as in a standard mixture-model, are drawn from a Dirichlet distribution for each example in the data. This re-sampling allows a single example, in this case a CT scan, to have been generated by a mixture of aspects.

We shall derive a new multilevel extension to LDA and apply this model to CT image data. The results act as a demonstration of the techniques involved, and as no hand segmented scans are available no absolute measure of the accuracy of results can be given.

The spectrum of lung disease naturally falls into a classification in the form of a hierarchy. Figure 3.1 was drawn up by a consultant radiologist. In the first level of the hierarchy the classification is most general, with the distinction of *More Dense* and *Less Dense* tissue. *More Dense* tissue will have a higher Hounsfield unit and appear a lighter colour on a CT image where as *Less Dense* tissue will appear darker. As you progress down the hierarchy the classifications become more specialised ending at the leaf nodes with well known diseases such as *Fibrosis* and *Emphysema*. We want try to recreate the hierarchy in an unsupervised

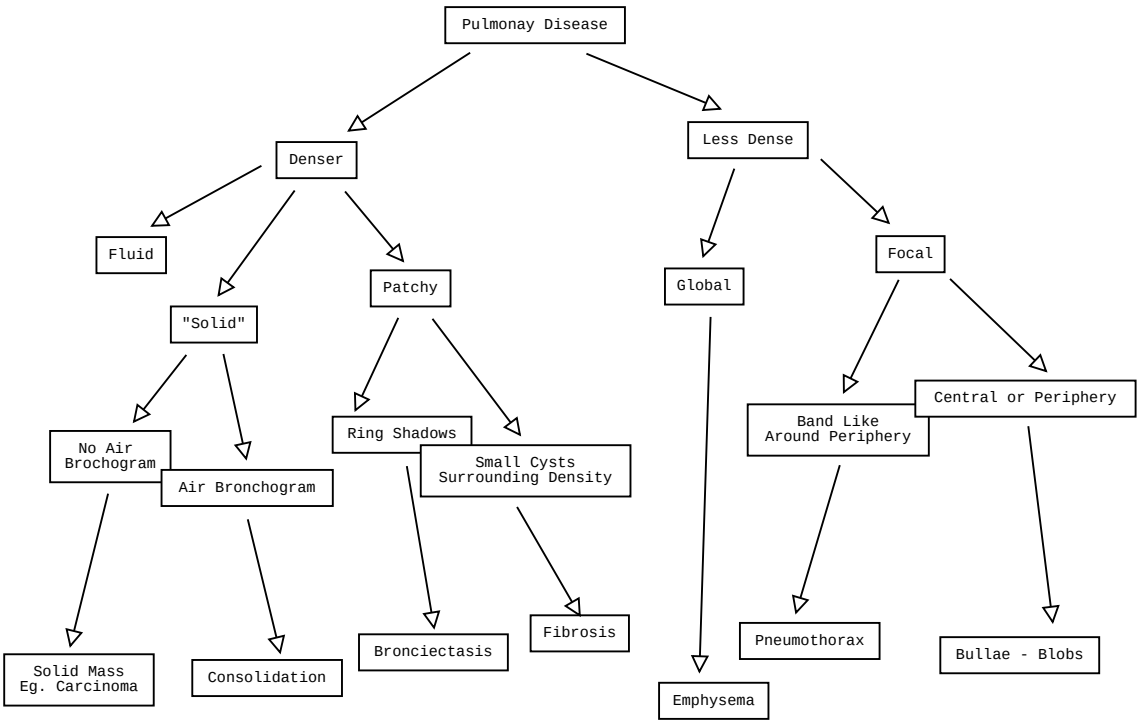


Fig. 3.1: Consultant's Hierarchy of Disease

way. As the top level only contains *Less Dense* and *More Dense* tissue, we would need to extend this to include *Normal* as well. So in total for our unsupervised approach we would be looking for 3 top level nodes and 7 lower level nodes.

3.3 Data and Methods

The CT scans are a collection (∼40 per person) of longitudinal cross sections taken from the chest. For more in depth discussion of CT chest scans see chapter 1. The lung area was segmented using simple thresholding and the remaining image split into 4×4 pixel regions. A set of eighteen statistical image features is then generated for each of these regions. Each feature is based on the intensity of the pixels.

These features were

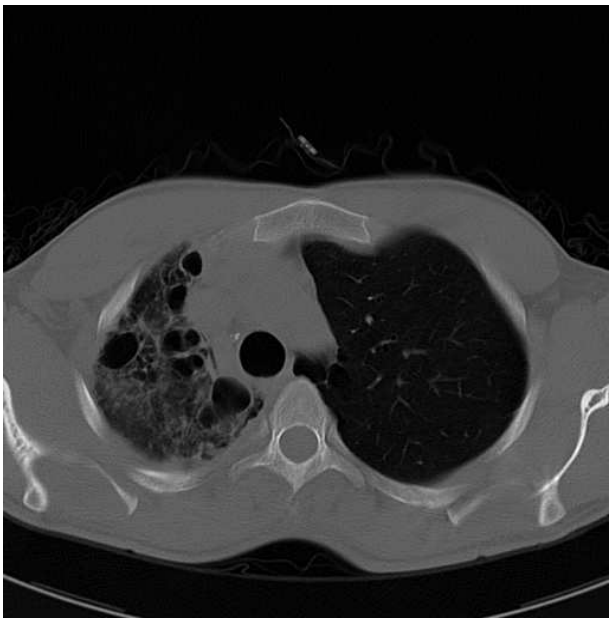


Fig. 3.2: Example Image

- Statistical Based: Mean, Maximum, Minimum, Range, Standard Deviation, Lower 25th percentile, Upper 75th percentile, Mean Average Deviation, Skewness, Kurtosis.
- Fourier Transform Based: Maximum, Average, Energy.
- Fractal Based: Fractal Dimension.
- Autocorrelation Base: First calculate the one step autocorrelation for 3 directions - horizontal, vertical and diagonal. Then take the Sum, Product, Maximum and Minimum of these three values.

The CT scan shown in figure (3.2) is a 512×512 pixel image given in Hounsfield units (-1000 for air +1000 for bone). This has been converted to a grey scale for viewing. To try and mirror the structure given in 3.1, a straight forward approach would be to cluster an individual scan using a hierarchical agglomerative approach. Then chop the tree to give 3 top level nodes and 7 lower level nodes. Hierarchical clustering with a Euclidean metric and *average link* merging was applied to the single image given in figure 3.2. For computational reasons simultaneous hierarchical clustering of more than one image was not possible. For this reason clustering results from this method will not give an comparable overview of the

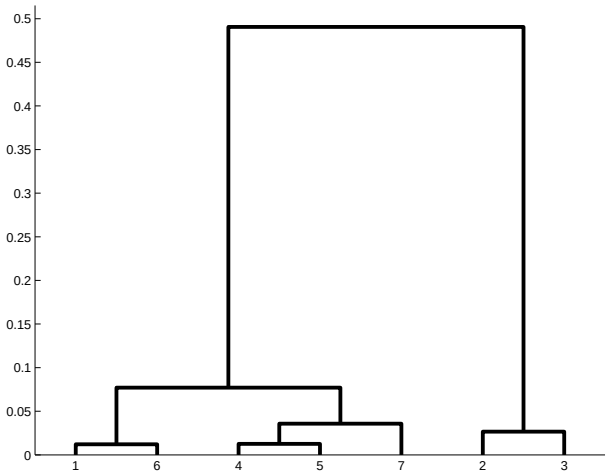


Fig. 3.3: Dendrogram for one scan

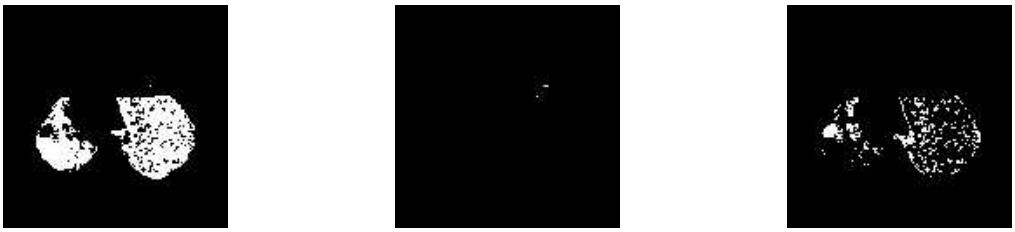


Fig. 3.4: Images for node 4,5 and 7

whole image database.

Figure 3.3 shows the derived hierarchy for a single image. While figures 3.3 and 3.5 show the decomposition of the image into the 7 leaf nodes, and figure 3.6 shows the top 3 parent nodes. In each image white indicates membership of that cluster. At the lower level, the clustering algorithm splits the image into seven clusters, three of which are significant. The cluster for node 4 given by the first image in figure 3.3 shows mainly *normal* and *less dense* tissue, while the cluster for node 7 given by the third image in figure 3.3 shows more dense tissue. The remaining cluster of significance is cluster for node 1. This is essentially modelling noise that is often present in a CT image at the boundary of the lungs.

As an alternative approach which would encompass the whole data set we constructed a probabilistic model which would cluster regions from individual CT images into a hierarchy. This hierarchy would be inferred using all the CT images available. The model is given in

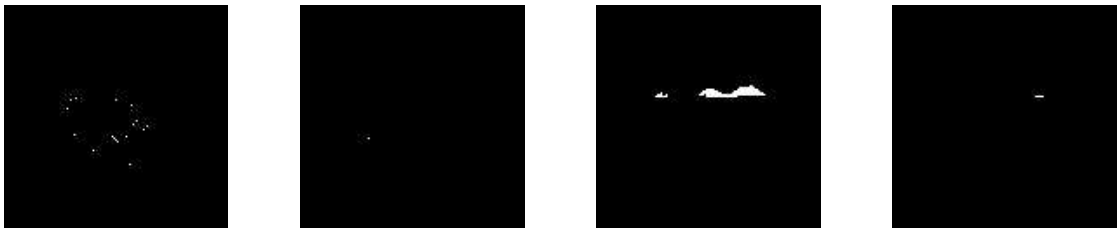


Fig. 3.5: Image for nodes 2 and 3 and for nodes 1 and 6

Fig. 3.6: Images for three parent nodes. Specifically, the parent of groups '4 + 5 + 7', '1 + 6' and '2 + 3' respectively.

figure (3.7).

One of the assumptions we make is that distinct tissue types, such as *Fibrotic*, *Normal* or any number of other possibilities, broadly fall into clusters that are defined by their features. This is reasonable as the first node in figure 3.1 separates *less dense* from *more dense* which is very closely related to the mean pixel value. Additionally the model assumes that each image is made up of a mixture of different *tissue types*. These *types* come from distinct Gaussian distributions, with a unique mean μ and variance σ for each of the 18 statistical features. The proportions of each *type* present is specific to an individual image and has been sampled from a Dirichlet distribution with parameter α.

There is also a second level which allows the division of the *types* into further *subtypes*. The *types* are related to the *sub-types* through a multinomial parameter β. As is standard for graphical models the circles in figure (3.7) give variables, with α, μ, σ, $\mu^{'}$ $\sigma^{'}$ and β model parameters to be estimated and θ, Z and Y latent variables. Arcs show conditional dependencies between variables and frames indicate ex-changeability. The generative process is such that the images are each generated twice. Once in the upper level of the hierarchy and once in the lower, the connection between these two levels is linked by the parameter β which is model parameter to be estimated. More formally the generative process for each

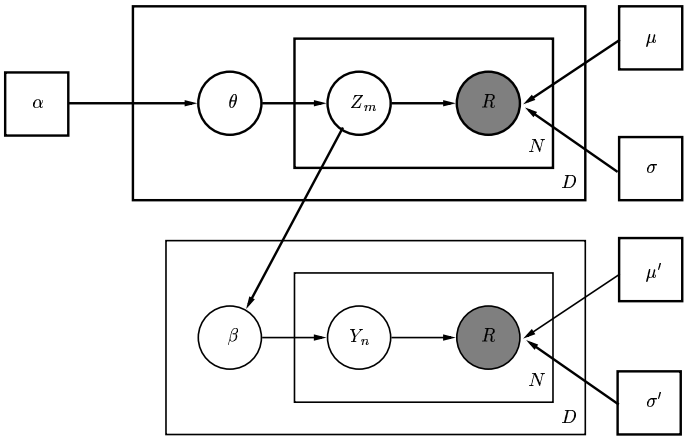


Fig. 3.7: Generative Model for the Hierarchical extension to LDA. The Shaded nodes indicate the image regions, the observed data. Square nodes in the model parameters indicate we are making a point estimate of these.

image is:

- Sample a multinomial θ from a Dirichlet distribution. This defines the mixing quantities at the upper level in the hierarchy.
- Top Level Image Generation:
- $\rightarrow$ For each region, R_{nd}, in the image draw a sample from θ to give a process $Z = k$. This Z then defines the set of Gaussian distributions for each feature for this region.
- $\rightarrow$ For each feature, R_{ndf}, for the region sample the chosen Gaussian. This has parameters μ_{fk} and σ^2_{fk}.
- Lower Level Image Generation:
- $\rightarrow$ For each region, R_{nd}, in the image draw a sample from $P(Y_n = k'|Z_n = k, \beta)P(Z_n = k)$ to give a process $Y = k'$. This Y then defines the set Gaussian distributions for each feature for this region.
- $\rightarrow$ For each feature, R_{ndf}, for the region, sample the chosen Gaussian. This has parameters $\mu^{'}_{fk'}$ and $\sigma^{'2}_{fk'}$.

The indices's read as follows: d is an image index, n is for the number of regions within

an image, f is for the features, k and $k^{'}$ index *types* and *subtypes* respectively. There are many ways to estimate unknown parameters for a probabilistic model [57]. We shall use Variational Expectation Maximisation (EM) to give us a point estimate of the posterior parameter distributions.

$$\begin{aligned} P(d|\mu,\sigma,\beta,\alpha,\mu^{'},\sigma^{'},\beta) = \quad & \int_{\triangle} \prod_{n}^{N_d} \sum_{k} P(R_{nd}|Z_n=k,\mu,\sigma)P(Z_n=k|\theta)P(\theta|\alpha) \\ & \prod_{n} \sum_{k,k'} P(R_{nd}|Y_n=k',\mu^{'},\sigma^{'}) \\ & P(Y_n=k'|Z_n=k,\beta)P(Z_n=k|R_{nd},\theta)d\theta \end{aligned} \tag{3.1}$$

The likelihood for the model is given by equation (3.1). The first part

$$\int_{\triangle} \prod_{n}^{N_d} \sum_{k} P(R_{nd}|Z_n=k,\mu,\sigma)P(Z_n=k|\theta)P(\theta|\alpha)d\theta$$

accounts for the upper plate in figure 3.7, the initial clustering stage. While

$$\prod_{n} \sum_{k,k'} P(R_{nd}|Y_n=k',\mu^{'},\sigma^{'})P(Y_n=k'|Z_n=k,\beta)P(Z_n=k|R_{nd},\theta)$$

accounts for the lower plate, namely the hierarchical structure and bottom level clustering.

To apply a Variational EM algorithm we have to first form a tractable lower bound on the likelihood in equation 3.1. We apply Jensen's inequality three times. First to break coupling between variables within an integral and then twice to break coupling between variables within a summation. This introduces three latent variables, γ a image specific Dirichlet parameter, ϕ the probability that in the first level of the hierarchy, region n in image d has been generated by process k and η which is analogous to ϕ for the second level. This bound is then maximised for all model parameters and latent variables. A full derivation is given in

appendix B. The resulting update equations are:

$$
\begin{aligned}
\mu_{fk} &= \frac{\sum_{d,n} \phi_{ndk} R_{ndf}}{\sum_{d'} \phi_{nd'k}} \\
\sigma^2_{fk} &= \frac{\sum_{d,n} \phi_{ndk} (R_{ndf} - \mu_{fk})^2}{\sum_{d'} \phi_{nd'k}} \\
\phi_{ndk} &\propto \prod_f P(R_{ndf} | Z_n = k, \mu_{fk}, \sigma_{fk}) . \exp\left[\Psi(\gamma_{dk}) - \Psi(\textstyle\sum_k \gamma_{dk})\right] \\
&\quad \times \exp \textstyle\sum_{f,k'} \eta_{ndk'} \log \left[P(R_{ndf} | Y_n = k', \mu^{'}, \sigma^{'}) \right] . \exp \textstyle\sum_{k'} \eta_{ndk'} \left[\log \beta_{kk'} + \log \eta_{ndk'}\right] \\
\mu^{'}_{fk'} &= \frac{\sum_{d,n} \eta_{ndk'} R_{ndf}}{\sum_{d'} \eta_{nd'k'}} \\
\sigma^{'2}_{fk'} &= \frac{\sum_{d,n} \eta_{ndk'} (R_{ndf} - \mu_{fk'})^2}{\sum_{d'} \eta_{nd'k'}} \\
\eta_{ndk'} &\propto \prod_f P(R_{ndf} | Y_n = k, \mu^{'}_{fk'}, \sigma^{'}_{fk'})) . \exp(\textstyle\sum_k \phi_{ndk} \log \beta_{kk'}) \\
\beta_{kk'} &\propto \textstyle\sum_{d,n} \phi_{ndk} \eta_{ndk'} \\
\gamma_{dk} &= \alpha_k + \textstyle\sum_n \phi_{ndk} \\
\alpha_{new} &= \alpha - H(\alpha)^{-1} g(\alpha)
\end{aligned}
\tag{3.2}
$$

Where H is the Hessian and g the gradient of the bounded likelihood with respect to the parameter α_k. Expressions for these are given in equations 3.3 and 3.4. Due to the special form of the Hessian, a diagonal + constant, inversion is straightforward (see [15] for details).

$$
g(\alpha) = \frac{\partial L}{\partial \alpha_i} = D \left[\Psi(\sum_k \alpha_k) - \Psi(\alpha_i) \right] + \sum_d \left[\Psi(\gamma_{di}) - \Psi(\sum_{k'} \gamma_{dk'}) \right] \tag{3.3}
$$

$$H(\alpha) = \frac{\partial L^2}{\partial \alpha_i \partial \alpha_j} = D\Psi^{'}\left(\sum_k \alpha_k\right) - D\Psi^{'}(\alpha_i)\delta_{ij} \tag{3.4}$$

The equations 3.2 are iterated until convergence. The approach is attractive as it is an intuitive way in which we can think about the structure of the lung. It assumes that there is a finite number of appearances or *themes* for each region in the lung. The number of *themes* is defined by the number of processes in the top level of the hierarchy. These general types can then be split into a number of sub-clusters (defined by the number of process in the second level) which represent more specific tissue types.

To calculate the likelihood we need to evaluate 3.1. As this contains an intractable integral an approximation technique is needed. It would be possible to appeal to the Monte Carlo framework from section 2.3.7, but we shall use a simple averaging method. This first requires drawing T samples, $\theta_1, \ldots, \theta_T$ from a Dirichlet distribution using the estimated model parameter α, and then evaluating the sum

$$\begin{aligned} P(d|\mu, \sigma, \beta, \alpha, \mu^{'}, \sigma^{'}, \beta) \simeq \quad & \tfrac{1}{T}\textstyle\sum_t \prod_n^{N_d} \sum_k P(R_{nd}|\mu_k, \sigma_k)\theta_{tk} \\ & \textstyle\prod_n \sum_{k,k'} P(R_{nd}|\mu^{'}_{k'}, \sigma^{'}_{k'})\beta_{kk'} P(Z_n = k|R_{nd}, \theta_t) \end{aligned} \tag{3.5}$$

3.3.1 Results and Comment

The model was applied to a collection of 60 individual CT images taken from a total of 24 patients. After segmentation, there were typically ∼4000 resulting 4×4 regions in each scan. To estimate the parameters update equations (3.2) would be then iterated until the parameters had converged. Although we know that we want to choose 3 processes in the upper level and 7 processes in the lower level of the hierarchy it is possible to optimally choose these by performing a cross validation. This is done by sequentially leaving out 10% of the data, then estimating the model parameters on the remaining 90%. The likelihood L_1 of the held out 10%, with respect to the estimated parameters, is then approximated using equation 3.5. This is then repeated for all 10 cross validations to give a mean log likelihood

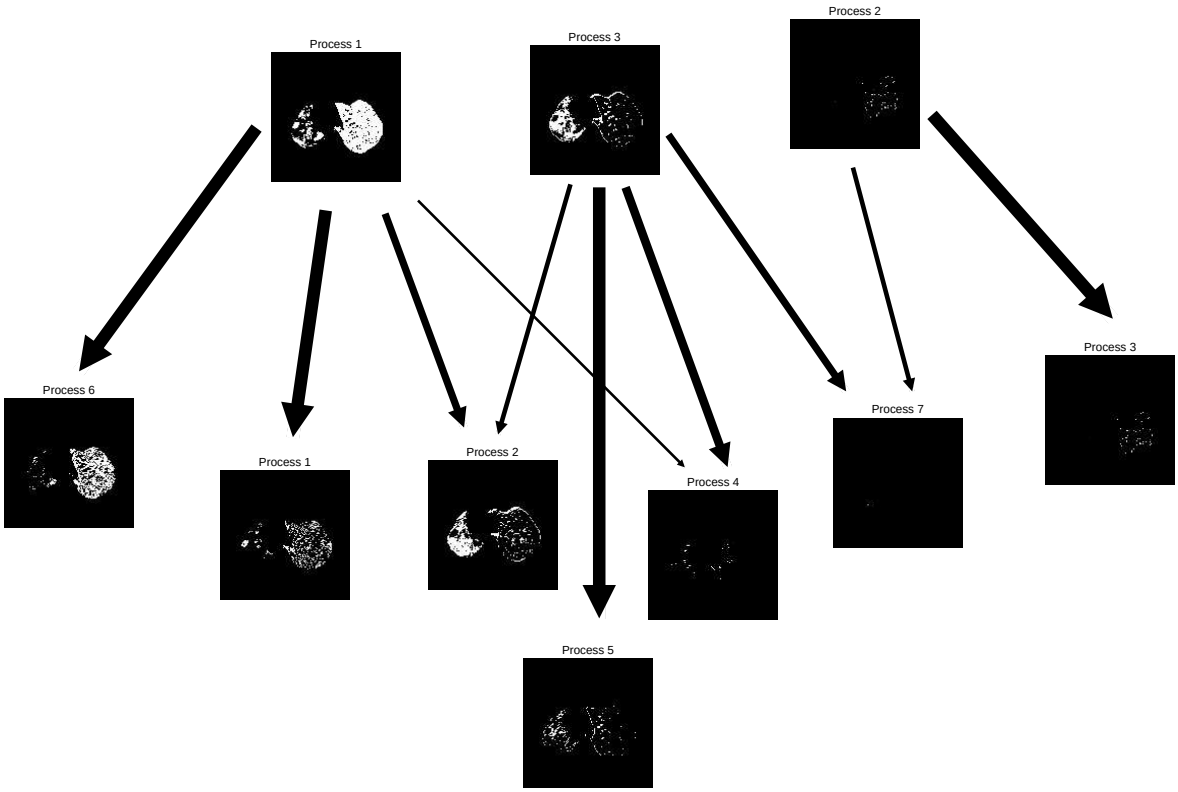


Fig. 3.8: Figure showing membership to each process in the hierarchical model. The width of an arrow is $\propto \beta$

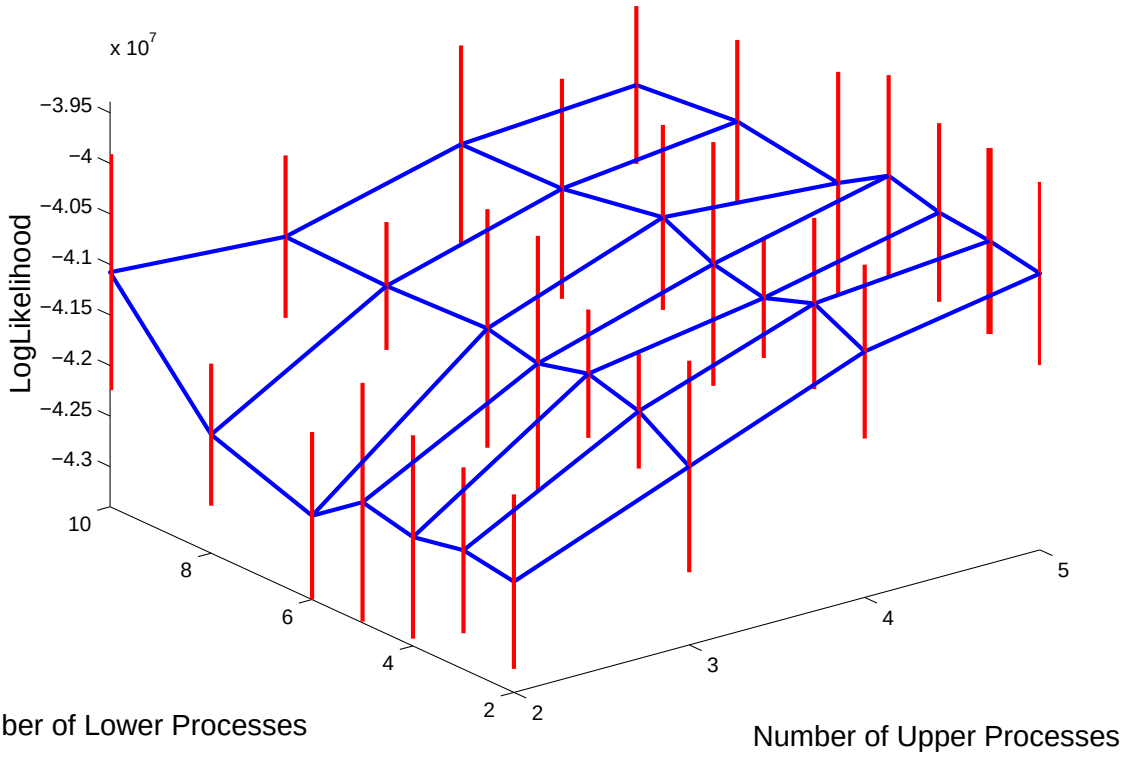


Fig. 3.9: A plots showing the average held out log likelihood with standard error bars for varying choices of the number of upper and lower processes in the hierarchy.

$$\hat{L} = \frac{1}{10} \sum_{i}^{10} L_i$$

and a standard error

$$E = \frac{1}{\sqrt{10}} \sum_{i}^{10} \sqrt{L_i^2 - \hat{L}^2}$$

A plot of the mean log likelihood for vary choices of the number of upper and lower processes is given in figure 3.9.

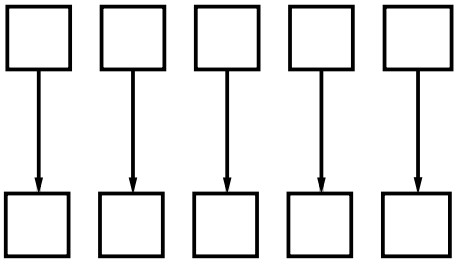


Fig. 3.10: A plot showing the likely resulting hierarchy for a choice of 5 processes in the upper and lower levels. Note the connecting parameters β would be 1 for one connection between processes and zero for all others

The maximum cross validated likelihood in figure 3.9 is obtained at (5,5). This is not a particularly significant result as the standard errors for each average are larger than the differences between these averages. However it is of no surprise that maximum is obtained when there are the same number of processes in the upper and lower levels of the hierarchy. In both levels of the hierarchy a number of processes is being fitted to the same data, so one would expect the optimal number to be the same in each case. With this choice it is likely that the connecting parameter β would give a 1 to 1 mapping between upper and lower processes. This is demonstrated in figure 3.10. Although the cross validated maximum likelihood solution is interesting, we will choose 3 processes in the upper level and 7 processes in the lower level of the hierarchy as this is what was given originally in consultants hierarchy of disease (shown in figure 3.1).

First we shall consider an image that was present in the data used to estimate the parameters. Figure 3.8 shows the decomposition of figure 3.2 into 3 top level processes and 7 sub processes. The intensity of the pixel shows the probability that that region was generated by the given process. This is simply the latent variable ϕ_{ndk}. The width of the connecting arrows are $\propto \beta$, a multinomial parameter with a range of 0-1, which sums to 1 for a given lower level process. 3 and 7 processes were chosen as this corresponded to the consultants original hierarchy. We see that in the top level process 1 represents all the Normal, all the Emphysema and a small amount of Mild Fibrosis, Process 2 is empty and Process 3 represents more severe Fibrosis. The lower level process 6 contains all Normal tissue and comes entirely from top process 1. Process 1 is all Emphysema and also comes from top process 1, and process 2 is all Fibrosis and contains the regions from top processes 1 and 3. The remaining Processes are all essentially empty for this image. It is interesting to note that Emphysema and Normal

Process	One	Two	Three
$P(Image\|Process)$	0.4222	0.4352	0.1425

Tab. 3.1: Probability of the image given in figure 3.11 for each process in the top level

Process	One	Two	Three	Four	Five	Six	Seven
$P(Image\|Process)$	0.0262	0.1447	0.3728	0.0133	0.0497	0.3553	0.0381

Tab. 3.2: Probability of the image given in figure 3.11 for each process in the lower level

were grouped together in the top level decomposition. This demonstrated greater difference between the regions represented by Process 2 and 3 than between Emphysema and Normal. As an example of classification we shall *classify* an image that was not present in the training data. To test an image we retain the model parameters $\mu, \sigma, \mu', \sigma', \alpha$ and β (β is a model parameter as it is external to the second level). The probabilities for the generation of each region by each process in both levels of the hierarchy are then calculated from the unseen data and model parameters. For the 4×4 testing image there are 3130 regions so only a normalised sum of these probabilities will be given across all the regions for each process. These normalised sums are given in tables 3.1 and 3.2. The unseen image to be classified into the hierarchy is given in figure 3.12. This patient is suffering from Fibrosis, shown as the denser areas. This is especially seen in upper section of the right lung. The large circular shape appearing in the left lung is the top of the liver and should be regarded as noise. Figure 3.12 gives the top level decomposition and figures 3.13 and 3.14 give the lower level decomposition. In the top level, process 2, which was essentially empty in figure 3.8, is all Normal tissue. This decomposes further in the second level giving lower level process 3. Process 1 shows a mixture of Fibrosis and Normal. This decomposes into lower level processes 2 and 6. Lower process 2 is all Fibrosis and lower process 6 is all Normal, which are both agreement with the same process in figure 3.8. Comparison with the image shown in figure 3.8 suggests that there is often a blurring of the lines between Emphysema, Fibrosis and Normal. A hierarchical decomposition gives us an idea of where we should introduce disease sub-types, and to what extent the standard medical classification into disease types is appropriate for machine learning problems.

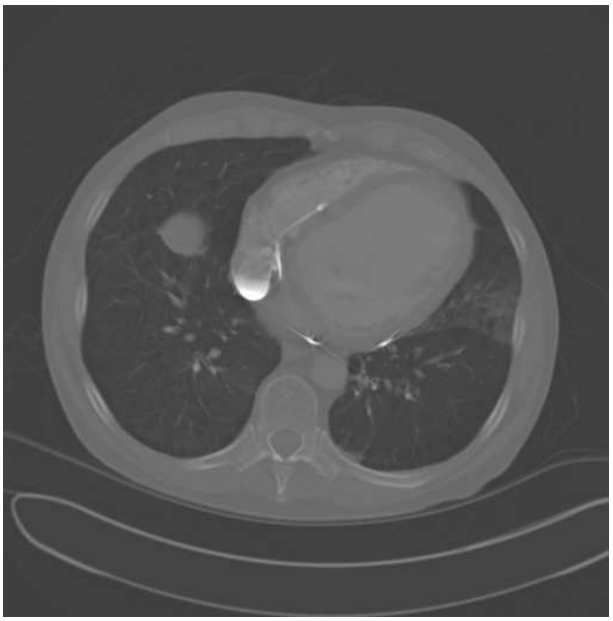


Fig. 3.11: Unseen image

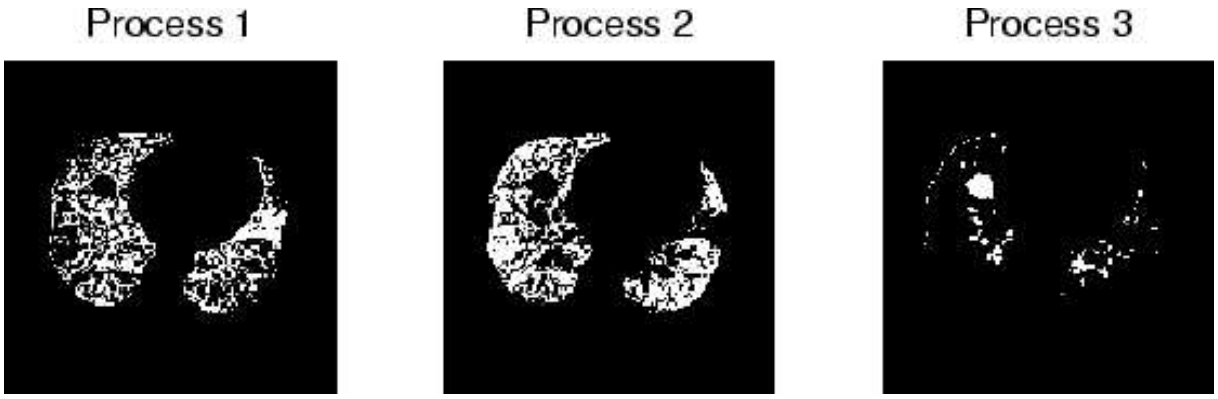


Fig. 3.12: Top Level decomposition

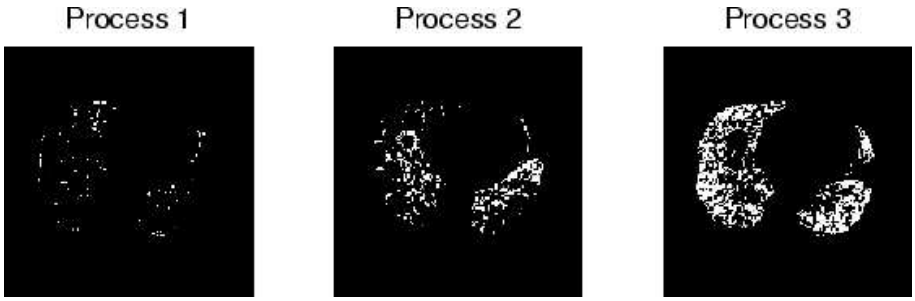


Fig. 3.13: Processes 1-3 in the lower level decomposition

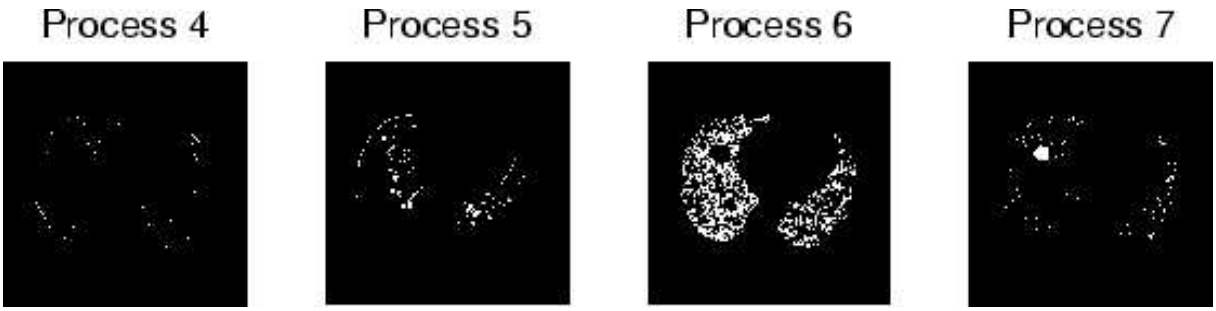


Fig. 3.14: Processes 4-7 in the lower level decomposition

3.4 Conclusions and Future Work

We have demonstrated a novel extension of LDA to radiological data. The estimated posterior can be used for prediction and classification. The whole process is completely unsupervised, this is an important point to make in the case of medical image data as hand labelled scans are expensive in time and often very dependent on the expert involved. Unsupervised methods can additionally highlight were specific disease classifications are unsuitable for a machine learning algorithm. An example of this is consolidation. This is described in [38] as *'when air within the acinus is replaced by fluid, tissue or exudate resulting in opacification of the parenchyma'*, it is a situation in which the finding can have a number of very distinct appearances. Without individual labelling of these sub-types a supervised algorithm would, without much success, try and learn the defining features of consolidation. Conversely an unsupervised algorithm could identify the sub-types and model each one individually with greater success. It is clear that automated radiological diagnosis is a complex problem. There are an immense number of possible diseases and widespread variability in their appearance. On a relatively small data set we have shown some promising results that would no doubt be improved given more examples. A natural progression from the models given in this paper has been to construct a framework that simultaneously models both the image data and textual report data [19]. Using this it is possible to learn to classify CT images into specific diseases in a completely unsupervised manner.

Chapter 4

Unsupervised Learning in Radiology using Novel Latent Variable Methods

4.1 Abstract

In this paper we compare a variety of unsupervised probabilistic models used to represent a data set consisting of textual and image information. We show that those based on Latent Dirichlet Allocation (LDA) outperform traditional mixture models in likelihood comparison. The data set is taken from radiology; a combination of medical images and consultants reports. The task of learning to classify individual tissue, or disease types, requires expert hand labelled data. This is both expensive to produce and prone to inconsistencies in labelling. Here we present methods that require no hand labelling and also automatically discover subtypes of disease. The learnt models can be used for both prediction and classification of new unseen data. This work was presented at and appeared in the proceeding of CVPR - *IEEE Computer Society International Conference on Computer Vision and Pattern Recognition 2005 - San Diego, CA, USA*, June 20-25, 2005, [19].

4.2 Introduction

Learning the relationship between images and textual data has been studied in a wide range of applications ([9], [8],[16]).The majority of these have been in the field of supervised classification, that is where the textual data contains only one element and so has a clearly defined correspondence with the image. It is common in radiology ([81]), to expertly label portions of a CT scan with its tissue type and then train a classifier using this labelled data to recognise unseen examples of labelled diseases. Though these methods can be on the whole very successful, they have a number drawbacks. The process is dependent on a substantial quantity of expertly labelled data, which is both expensive to generate and can often contain internal inconsistencies. In ([76]), great care is taken in deciding which regions of interest (ROIs) are used, and what proportion of a ROI can in-fact be considered truly representative of disease. Another drawback in labelling is that most disease cannot be diagnosed by taking an image region in isolation, but is correctly classified using other additional information such as patient history and symptoms. There is also no guarantee that a well established medical classification of disease should necessarily translate to a well defined pattern recognition problem. Here we are interested in demonstrating advantages of Computer Aided Diagnosis, and exploiting these, not trying to create an *Artificial Radiologist*. To avoid the problem of labelling individual regions we use entirely unsupervised methods. On one level, the models used here can be seen as a clustering of fused data. The data consists of Radiology reports and axial Computed Tomography (CT) chest scans. The data is linked by patient number so that a report can be paired with its corresponding images. The Radiology reports are free-text documents describing history, symptoms and observations. For the purposes of this experiment the reports are converted into binary vectors of the words of interest - *Fibrosis, Normal, Emphysema*. Additional words such as *Mild/Early Fibrosis, Ground Glass, Consolidation* were considered, but as we are in part attempting to discover classifiable sub types of disease we take only the most general terms. These words can be extracted by key-word searching, or by using the specialist Natural Language Processor MedLEE ([35]). The CT scans are a collection ($\sim$40 per person) of longitudinal cross sections taken from the chest. Figure (4.5) is given as an example, it is a 512×512 pixel image given in Hounsfield units (-1000 for air +1000 for bone), and is converted to a grey scale here for viewing. The lung area is segmented using a simple threshold and region growing algorithm and then split into square blocks. A set of eighteen image features is then generated for each block. Blocks

are taken to be of size 4×4, 8×8 or 16×16 pixels, giving three separate data sets. The image features are transformed to have zero mean and unit variance. In total there were 310 individual CT scans, taken from a population of 24 patients. If expertly labelled training data does exist, then this can easily be incorporated into the model as a *Faked* image-report pairing in which there exists only a single region and a single term. In this study no such data was used. Feature selection for classification of CT chest images has been studied in great detail ([82],[76]). In this paper we restrict ourselves to common statistical features such as moments, autocorrelations and FFTs. The probabilistic models given here are all generative mixtures. That is, the data is assumed to have been generated under a scheme containing a mixture of classes. We shall refer to these classes as processes or *topics*. Latent variable methods work by assuming that some variables of interest are missing. Their values are then estimated based on the observed data. The introduced latent variables often give tractability to parts of the problem. The simplest model discussed here is a Gaussian-Multinomial mixture model. Mixture model approaches have been used before in medical applications for the segmentation of images([64]). For the problem of data fusion standard mixture models are very restrictive. They require both data elements, in this case all terms and regions in an image, to have been generated by the same underlying process. The richer class of models we use that are based on LDA allow a re-sampling of distributions for individual regions (or even features) within an image. This is essential in the analysis of chest images as disease can always be localised within a small area, and almost never has a global (entire image) appearance. Models combining words and images in an LDA based generative model have been introduced in [8] and [16].

4.3 Models and Methods

We applied a selection of probabilistic models to the data. Two assumptions are made across all models, the first is that the statistical image features can be represented by some combination of Gaussian, $\mathcal{N}(\mu, \sigma)$, distributions, and the second that the distribution of report words forms a Multinomial, $Mult(\beta)$. We shall denote D as the data which is a CT scan combined with a radiology report. In each of the following models $\mathcal{L} = P(D|\Theta)$ is the likelihood of this combined data given the model parameters Θ. In all the equations that follow the notation is consistent.

- $P(R_{ndf})$ is the probability of feature f, in region n in document d. The n and f are sometimes omitted to indicate a product over features and regions.

- μ is the mean of a normal distribution. This has an index over features, f and processes k.

- σ^2 is the variance of a normal distribution. This also has an index over features, f and processes k.

- Multinomial distributions over the corpus are given by β. This is indexed by word m and process k.

- K and M are taken as indexes for processes and terms respectively.

- Ψ is used to denote the derivative of the log gamma function $\frac{d}{dz}log(\Gamma(z))$.

In total we study five models. One model is based on a traditional mixture, while the remainder all incorporate LDA re-sampling.

4.3.1 Mixture Models

The first model is the simplest of all. In this the multinomial and Gaussian distributions are jointly drawn under a mixture. This means that each Report-Image pair contains only one *topic*. This may seem very restrictive, but when using probabilistic models it is important to consider the simplest approaches to identify model over-fitting. The likelihood for this model is given in equation (4.1).

$$\mathcal{L} = \sum_k \alpha_k \prod_m P(W_m|\beta_k) \prod_n P(R_n|\mu_k, \sigma_k) \tag{4.1}$$

Using a standard variational approach, a bound on the log-likelihood if formed using Jensen's inequality over the expectation of the latent variable γ_{dk}. This is given in equation 4.2:

$$\begin{aligned}\log \mathcal{L} &= \textstyle\sum_{dkm} \gamma_{dk} \log P(W_m|\beta_k) \\ &+ \textstyle\sum_{dkn} \gamma_{dk} \log P(R_n|\mu_k, \sigma_k) \\ &- \textstyle\sum_{dk} \gamma_{dk} \log \gamma_{dk} \\ &+ \textstyle\sum_{dk} \gamma_{dk} \log \alpha_k\end{aligned} \tag{4.2}$$

γ_{dk} is a parameter of a discrete variational distribution and taken as the probability that sample d was generated by mixture distribution k. This is very similar to the EM algorithm for a mixture model given in section 2.4.1. This bound on the likelihood is then maximised for all model and latent variables.

The remainder of the models are based on LDA. We shall only give brief details of the equations for the first three models as similar derivations exist elsewhere ([15], [16]), but as the *Reversed Correspondence-LDA* is new we shall give a more thorough explanation.

4.3.2 Joint-LDA

The next model is a joint Gaussian-Multinomial LDA. This is similar to *Mixture*, but it is more flexible by allowing each Region and Word to have been generated separately. For each sample, that is image report pair, a k-dimensional multinomial is drawn from a Dirichlet, $D(\theta|\alpha)$. Then for each region and every word, θ is sampled to give a process Z_n for a region and Z_m for a word. The corresponding Region or Word is then generated conditioned on $Z_{n/m}$. Thus every region and every word can come from any one of k distinct classes. The likelihood of a single is given in equation (4.3).

$$\begin{aligned}\mathcal{L} = \int_{\triangle} \textstyle\prod_m \sum_k \quad & P(W_m|Z_m = k, \beta) P(Z_m = k|\theta) \\ \textstyle\prod_n \sum_k \quad & P(R_n|Z_n = k, \mu, \sigma) \\ \times \quad & P(Z_n = k|\theta) P(\theta|\alpha) d\theta\end{aligned} \tag{4.3}$$

Variational inference can again be used to estimate the model parameters. To construct a tractable bound on 4.3 we will apply Jensen's inequality twice to a logged version of the likelihood. Applying it once will bring the log through the integral and introduce the sample specific latent variable γ_{dk}. $P(\theta|\gamma)$ is a variational distribution that has a sample specific Dirichlet parameter γ_d. For each sample, d, γ_d is a vector of k parameters. The log will now effectively decouple the elements of the likelihood containing W_m and R_n. Applying Jensen's inequality for a second time the log will be brought into the summation over k to break the coupling between the θ. This introduces the two latent variables ϕ_{ndk} and ψ_{mdk}. In both cases it is a discrete distribution. ϕ_{ndk} is interpreted as the probability that for sample d region n's features, f, were generated by the Gaussian distribution defined by process k. Correspondingly ψ_{mdk} is interpreted as the probability that in report d word m was generated by the Multinomial distribution defined by process k, β_{mk}.

The full bound on the likelihood taken over all sample is given by:

$$\begin{aligned}
\textstyle\sum_d \log[\mathcal{L}] \geq & \textstyle\sum_{d,n,k} \phi_{ndk} \log\left[P(R_{nd}|\mu_k, \sigma_k^2)\right] \\
+ & \textstyle\int_\Theta \sum_{d,n,k} \phi_{ndk} P(\theta|\gamma_d) \log\left[\theta_k\right] d\theta \\
- & \textstyle\sum_{d,n,k} \phi_{ndk} \log\left[\phi_{ndk}\right] \\
+ & \textstyle\sum_{d,m,k} \psi_{mdk} \log\left[\beta_{mk}\right] \\
+ & \textstyle\int_\Theta \sum_{d,m,k} \psi_{mdk} P(\theta|\gamma_d) \log\left[\theta_k\right] d\theta \\
- & \textstyle\sum_{d,m,k} \psi_{mdk} \log\left[\psi_{mdk}\right] \\
+ & \textstyle\int_\Theta \sum_d P(\theta|\gamma_d) \log\left[P(\theta|\alpha)\right] d\theta \\
- & \textstyle\int_\Theta \sum_d P(\theta|\gamma_d) \log\left[P(\theta|\gamma_d)\right] d\theta
\end{aligned} \tag{4.4}$$

By maximising equation 4.4 with respect to all the variables the resulting iterative scheme can be used by find good estimates of the model parameters.

1	2	1	1	3
1	2	3	1	3
1	3	1	1	1
1	2	2	2	1
1	1	2	1	1

Tab. 4.1: Example image with the generating process for each region shown

4.3.3 Correspondence-LDA Model

This follows the generative model first given in ([16]). It is an extension of other earlier LDA models. The idea is that first the image features are generated under a re-sampling model, then the report terms are sampled using the processes that contributed to image generation. Hence there is a correspondence between the term generation and image generation. A good analogy can be given by reference to table (4.1).

This shows a hypothetical image that has been generated by first sampling a $Dirichlet(\alpha)$ to give a multinomial θ, and then sampling from this multinomial for each region. The number is each region gives the corresponding chosen k, the generating process. To generate the M terms, you throw a dart at the image M times. At each throw the dart lands in a region $Y_m = n$, this region has a process number $Z_n = k$. A term is then generated by sampling the corresponding multinomial defined by Z_n . Thus for popular processes such as 1 (15 out of 25) you are more likely to get terms coming from multinomial β_k. The likelihood is given in equation (4.5). Variational inference can be again used to optimise a bound on equation (4.5). This introduces three sample specific latent variables, γ, ϕ_{nk} as given before and additionally λ_{nm}. λ_{nm} is the probability word m was generated after selecting the process which generated region n.

$$\begin{aligned} \mathcal{L} = \quad & \textstyle\int_\Theta \prod_n \sum_k \quad P(R_{nd}|Z_n = k, \mu, \sigma)P(Z_n = k|\theta) \\ & \textstyle\prod_m \sum_{k,n} \quad P(Y_m = n|N)P(Z_n = k|R_{nd}, \theta) \\ & \times \quad P(W_m|Y_m = n, Z_n = k, \beta)P(\theta|\alpha)d\theta \end{aligned} \tag{4.5}$$

Through the introduction of the latent variables, a bound on the log of equation 4.5 is given by:

$$\begin{aligned}
\textstyle\sum_d \log[\mathcal{L}] \geq & \textstyle\sum_{d,n,k} \phi_{ndk} \log\left[P(R_{nd}|\mu_k, \sigma_k^2)\right] \\
+ & \textstyle\int_\Theta \sum_{d,n,k} \phi_{ndk} P(\theta|\gamma_d) \log\left[\theta_k\right] d\theta \\
+ & \textstyle\int_\Theta \sum_d P(\theta|\gamma_d) \log\left[P(\theta|\alpha)\right] d\theta \\
- & \textstyle\sum_{d,n,k} \phi_{ndk} \log\left[\phi_{ndk}\right] \\
- & \textstyle\int_\Theta \sum_d P(\theta|\gamma_d) \log\left[P(\theta|\gamma_d)\right] d\theta \\
+ & \textstyle\sum_{d,k,m} \psi_{mdk} \lambda_{nmd} \log\left[P(W_d|Y_n = m, Z_m = k, \beta)\right] \\
- & \textstyle\sum_{d,n,m} \lambda_{nmd} \log\left[\lambda_{nmd}\right]
\end{aligned} \tag{4.6}$$

Again, by maximising equation 4.6 with respect to all the variables the resulting iterative scheme can be used by find good estimates of the model parameters.

4.3.4 Correspondence-LDA with re-sampling feature wise

The likelihood equation (4.7) is identical to that of the first Correspondence Model equation (4.5), except the order of feature generation ($\prod_f$) and the process selection ($\sum_k$) have been reversed. Consequently the latent variables λ and ϕ are four dimensional, the additional dimension being over features. This does offer a greater flexibility with re-sampling over regions and features, but with the downside of many additional parameters to estimate.

$$\begin{aligned}\mathcal{L} = \quad & \int_\Theta \prod_{n,f} \sum_k \quad P(R_{ndf}|Z_{nf}=k,\mu,\sigma)P(Z_{nf}=k|\theta) \\ & \prod_m \sum_{k,n,f} \quad P(Y_m = nf|N,F) \\ & \times \quad P(W_m|Y_m=nf, Z_{nf}=k,\beta) \\ & \times \quad P(Z_{nf}=k|R_{ndf},\theta)P(\theta|\alpha)d\theta\end{aligned} \tag{4.7}$$

4.3.5 Reversed Correspondence-LDA

This is a new and previously unreported model. It is motivated by the *Corr-LDA*, except in this model the terms form the defining part of the data and the images are generated from the terms. This is a reversal of the original correspondence LDA [16]. In addition there is a multinomial ν introduced to account for a limited vocabulary. The graphical representation shown in figure (4.1) summarises the generative process. Individual variables are explained below.

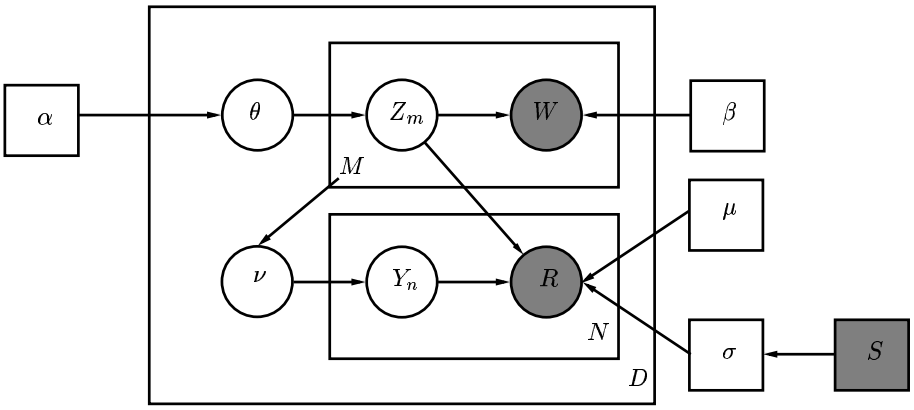


Fig. 4.1: Generative model for *Rev-LDA*. Note the fixed prior $\boldsymbol{S}$ on the variances is shaded to indicate this is static throughout the inference. Point estimates are given for all variables with square boxes.

In words the generative model is:

- Sample $\theta \sim Dirichlet(\theta|\alpha)$

- For each of the M words in a report

  1. Sample a process $Z_m \sim Multinomial(\theta)$
  2. Sample a word W_m from a multinomial over the vocabulary conditioned on the process.

- For each of the N image regions

  1. Pick a report word Y_n over a multinomial, ν (conditioned on that report).
  2. Sample the image features from Gaussian distributions conditioned on the process that was used to generate the word chosen, Y_n.

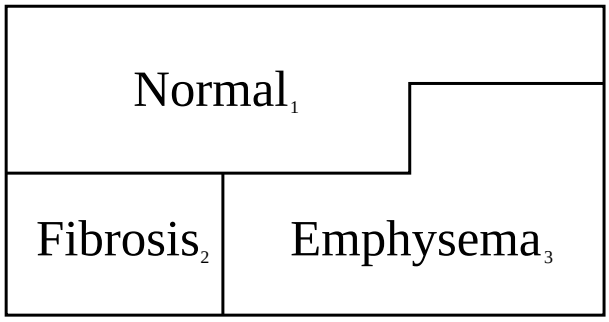


Fig. 4.2: Example Report

For a similar analogy to that given for the *Corr-LDA*, consider the *toy* report given in figure (4.2). This shows a hypothetical report where all terms (not always the case) have been sampled. The term subscript indicated the generating processes for that particular term, and the report specific multinomial ν then dictates the size of the term areas. To generate the image you would repeatedly throw darts at the report. At each throw a process $Z_m = k$ is determined by the subscript of the term area where the dart lands. This k is then taken as the process which generates the current image region. Thus the larger the term area the more image regions will be generated by that terms process. The likelihood of this model is given by equation (4.8).

$$\begin{aligned}\mathcal{L} = \quad & \int_\Theta \prod_m \sum_k \quad P(W_m|Z_m = k, \beta)P(Z_m = k|\theta) \\ & \prod_n \sum_{k,m} \quad P(Y_n = m|\mathbf{W}_d, \nu) \\ & \times \quad P(R_n|Y_n = m, Z_m = k, \mu, \sigma) \\ & \times \quad P(Z_m = k|W_m, \theta)P(\theta|\alpha)d\theta\end{aligned} \tag{4.8}$$

We have now detailed 5 alternative models by which the image and report data could have been generated. To compare these models we need to first estimate the model parameters for each. In terms of the total number of parameters the 5 models can be ordered as such:

- Mixture
- Joint-LDA
- Correspondence-LDA
- Reversed Correspondence-LDA
- Correspondence-LDA with re-sampling feature wise

Once we have estimated the model parameters we must then evaluate the likelihood. For example in the *Reversed Correspondence-LDA* model this requires integration of equation 4.8. As was discussed in section 2.3.7 this is seldom possible analytically.

To perform the parameter estimation in all the above models we shall use a variational EM approach. To recap from section 2.3.2 the variational EM approach will give a point estimate of the posterior distribution. By using Jensen's inequality of section 2.1.4, $f(E[x]) \geq E[f(x)]$, over the latent variables, we can give a lower bound on the integral that can be evaluated. This is then maximised over the model parameters. For the total data log-likelihood in the *Rev-LDA* model this bound is given in (4.9). The latent and model parameters are then iteratively updated by using the following equations. This is a kind of Expectation

Maximisation algorithm. There is no closed form update for α so a second order Newtonian method is used.

$$\begin{aligned}
\textstyle\sum_d \log[\mathcal{L}] \geq & \textstyle\sum_{d,m,k} \psi_{mdk} \log[\beta_{mk}] \\
+ & \textstyle\int_\Theta \sum_{d,m,k} \psi_{mdk} P(\theta|\gamma_d) \log[\theta_k]\, d\theta \\
+ & \textstyle\int_\Theta \sum_d P(\theta|\gamma_d) \log[P(\theta|\alpha)]\, d\theta \\
- & \textstyle\sum_{d,m,k} \psi_{mdk} \log[\psi_{mdk}] \\
- & \textstyle\int_\Theta \sum_d P(\theta|\gamma_d) \log[P(\theta|\gamma_d)]\, d\theta \\
+ & \textstyle\sum_{d,n,m} \lambda_{nmd} \log[P(Y_n = m|\nu_d)] \\
+ & \textstyle\sum_{d,k,m} \psi_{mdk}\lambda_{nmd} \log[P(R_d|Y_n = m, Z_m = k, \mu, \sigma)] \\
- & \textstyle\sum_{d,n,m} \lambda_{nmd} \log[\lambda_{nmd}]
\end{aligned} \tag{4.9}$$

E-Step:

$$\begin{aligned}
\psi_{mdk} \propto & \ \beta_{mk} \exp(\Psi(\gamma_{dk}) - \Psi(\textstyle\sum_k \gamma_{dk})) \\
& \exp(\textstyle\sum_n \lambda_{nmd} \log[P(R_{nd}|Y_n = m, Z_m = k, \mu, \sigma)]) \\
\lambda_{nmd} \propto & \ P(Y_n = m|\mathbf{W}_d) \\
& \times \exp(\textstyle\sum_k \psi_{mdk} \log[P(R_{nd}|Y_n = m, Z_m = k, \mu, \sigma)]) \\
\gamma_{dk} = & \ \alpha_k + \textstyle\sum_m \psi_{mdk}
\end{aligned} \tag{4.10}$$

M-Step:

$$\begin{aligned}
\mu_{fk} &= \frac{\sum_{d,n,m} \psi_{mdk}\lambda_{nmd}R_{ndf}}{\sum_{d,n,m} \psi_{mdk}\lambda_{nmd}} \\
\sigma^2_{fk} &= \frac{\sum_{d,n,m} \psi_{mdk}\lambda_{nmd}(R_{ndf}-\mu_{fk})^2}{\sum_{d,n,m} \psi_{mdk}\lambda_{nmd}} \\
\beta_{mk} &\propto \sum_d \psi_{mdk} \\
\nu_{dm} &\propto \sum_n \lambda_{nmd} \\
\alpha_{new} &= \alpha - H(\alpha)^{-1}g(\alpha).
\end{aligned} \tag{4.11}$$

4.3.6 MAP Solution

Solving for the *Maximum a posteriori* $P(\Theta|D) \propto P(D|\Theta)P(\Theta)$ problem we introduce priors over the model parameters. MAP solutions are scale dependent, but as we have normed the image data to have zero mean and unit variance an identical prior can be used for all the features. The most general prior over the variance which maintains an easy quadratic form for the update of σ is:

$$P(\sigma) = \frac{1}{Z(\sigma)} \exp(s_1/\sigma^2) \exp(s_2/\sigma)\sigma^{s_3}$$

With one of $s_{1/2} = 0$, this is in fact a restatement of the Gamma distribution. With a negative choice of s_1 and s_2, this gives bounded values, asymptotic to unity and passing through the origin. A negative choice of s_3 changes the asymptote to zero, and gives a convergent integral, hence proper prior. This prior will penalise under fitting (small σ) and over fitting (large σ) in the model. The form of $Z(\sigma)$ is complex but does not need evaluating in any of the calculations. In all the models we used the prior corresponding to $s_1 = s_2 = s_3 = -1$, this is by no means optimal and indeed could be optimised by a cross-validation of the likelihoods. For the model parameter β we use the empirical Bayesian smoothing outlined in ([16]). Priors for α and μ are assumed uniform. It is interesting to point out that for a single process, all of the above models are in-fact identical. In generative terms, for a single example, the

differences between models are summarised as:

- *Mixture.* The Image/Report pair comes from a single process.
- *Joint-LDA.* All regions and terms can come from any process - linked by a multinomial.
- *Corr-LDA.* Regions can come from any process - terms come from a selection of those generating the image.
- *Rev-LDA.* Terms can come from any process - regions come from a selection of those generating the report.

4.4 Results

To compare models a hold out cross validation was performed. This involves retaining a subset of the data, running the update equations on the remaining data until convergence, and then calculating the average held-out sample log-likelihood. To remove any issues of dependence between data the held-out samples were chosen to be those from an individual patient. So for each model and number of processes there was a 24 fold cross validation. A plot of the results for 4×4 and 16×16 blocks can be seen in figures(4.3-4.4). The likelihood values for the smaller blocks are taken across ~ 16 times as many regions which explains the difference in magnitude. As can be seen *Joint-LDA* and *Corr-LDA* significantly out perform the other models. *Corr-LDA-Feat* seems too complex and will over fit, where as *LDA-Rev* is too restrictive as the ratio of terms to regions is very low. The Mixture is too basic, only allowing one process per sample. We shall present results from the *Corr-LDA* as it is a richer model than *Joint-LDA*.

A demonstration of the results will be given with reference to figure (4.5). This patient is suffering from severe lung disease, there is a large area of fibrosis (denser tissue) in the right lung interspersed with patches of Emphysema (essentially air-sacs, very dark in appearance). This image was removed from the training set and classified (by process) using the model parameters. Figures (4.6-4.7) show the decomposition of figure (4.5) into 8 processes, although 8 processes is not the optimal number it is sufficiently high to demonstrate the results.

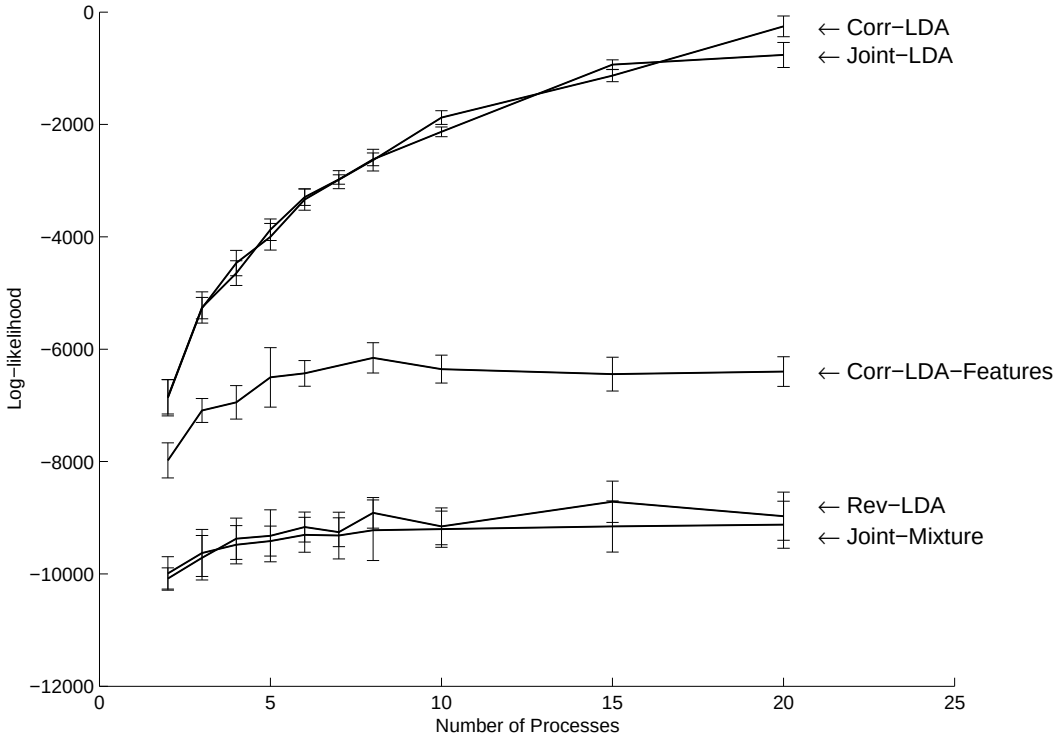


Fig. 4.3: Comparison of MAP Log-Likelihoods for different models and 4×4 region size

Corresponding to this decomposition are the multinomial probabilities over words given in table 4.2. First consider k=3 and k=6, from table 4.2 we see that these are principally *Emphysema*, in figure 4.6 we see that for k=3 this matches the emphysematous areas in the original CT scan very well. *Fibrosis* is demonstrated by k=4 and k=5 and to a lesser extent k=2. There is a very interesting decomposition between k=4 and k=5, while both being fibrotic k=4 would be classed as *Ground Glass*, a subtype of *Fibrosis*. The remaining processes account for the normal tissue. For comparison figure 4.8 shows the results of supervised classification for a CT scan. A selection of expertly labelled data was used to train a Support Vector Machine with a linear kernel. There is a clear correspondence between the results of the supervised SVM method, with the results of unsupervised learning given in this paper.

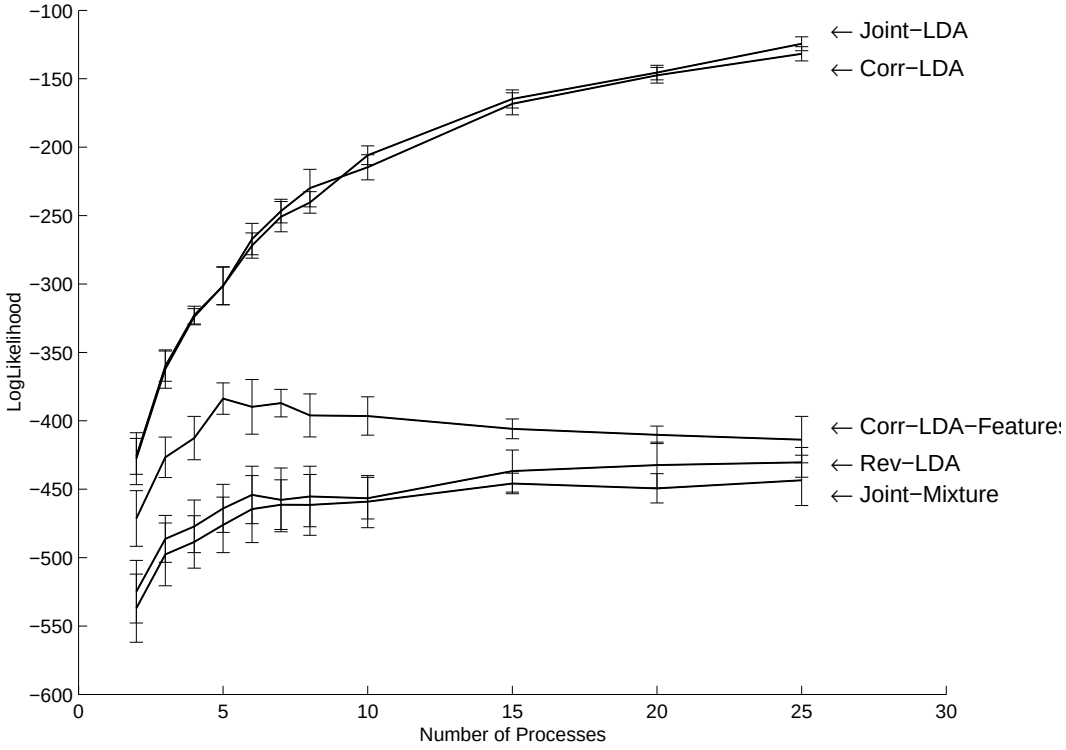


Fig. 4.4: Comparison of MAP Log-Likelihoods for different models and 16×16 region size

4.5 Conclusions and future work

In an entirely unsupervised manner, we have identified sub-types of lung disease and maintained a correspondence between these and established classes. This gives clearer indications of classes to use for automated diagnosis of disease. Note that we are estimating the parameters of the posterior distributions using the a variational method. At each stage when we apply Jensen's inequality we are stretching the the bound on the true likelihood. The algorithm only converges to local minima based on the bound, so we have no measure of the universal optimality of our solution. Other techniques, such as Monte Carlo methods or Expectation Propagation ([62]), exist and using these may provide superior solutions to those given above.

Process	Normal	Fibrosis	Emphysema
k=1	**0.860895**	0.13816	0.000945312
k=2	**0.622918**	**0.375928**	0.00115402
k=3	0.099295	0.00015781	**0.900547**
k=4	0.115884	**0.884013**	0.000103796
k=5	0.111384	**0.888615**	2.81294e-07
k=6	0.161668	2.69409e-07	**0.838332**
k=7	**0.848397**	0.000679466	0.150924
k=8	**0.991839**	0.000103146	0.0080576

Tab. 4.2: Table of smoothed β_{mk} for 8 processes in the 4x4 block data set, using the *Corr-LDA* model. Significant probabilities are shown in bold.

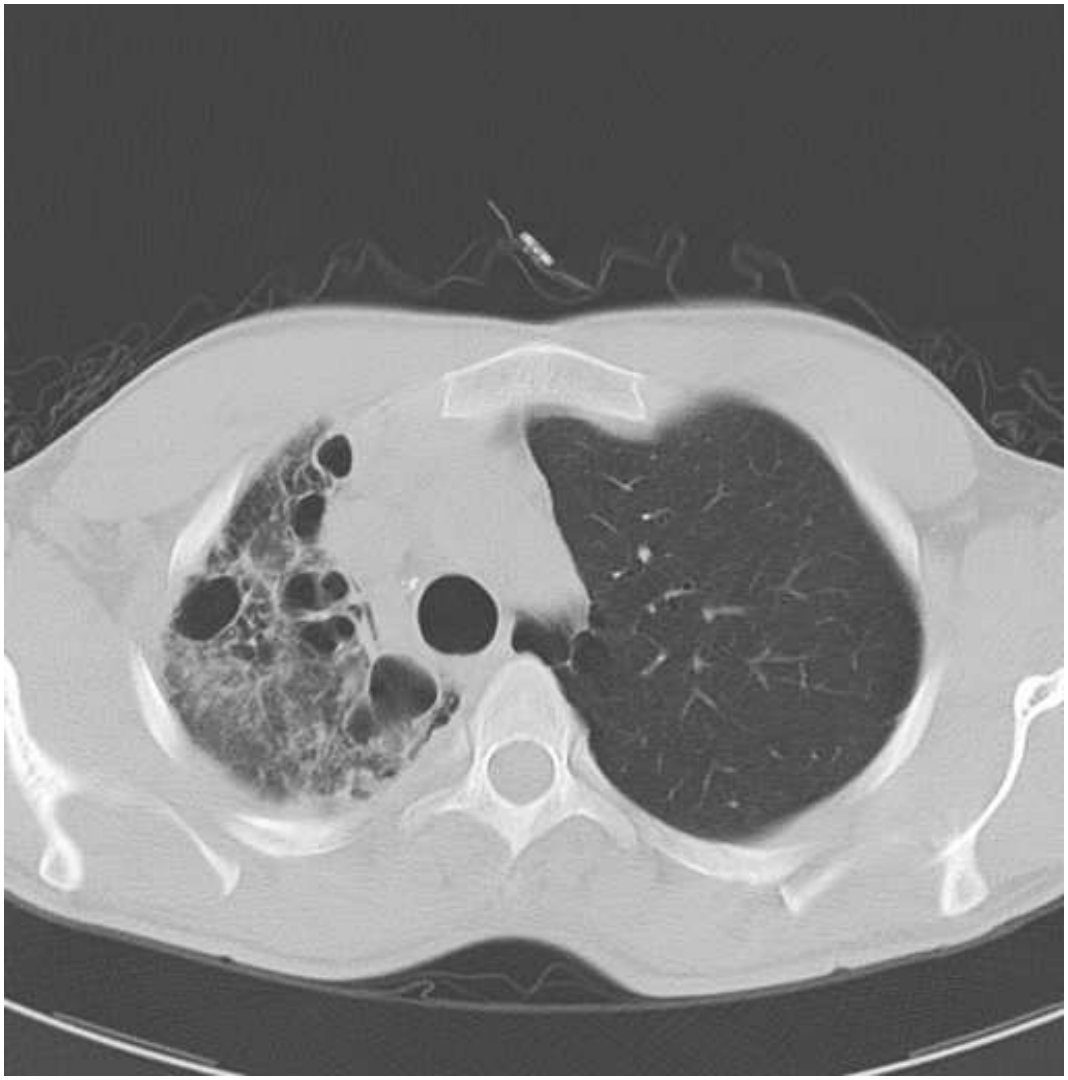


Fig. 4.5: Original CT Scan, Right/Left lung convention.

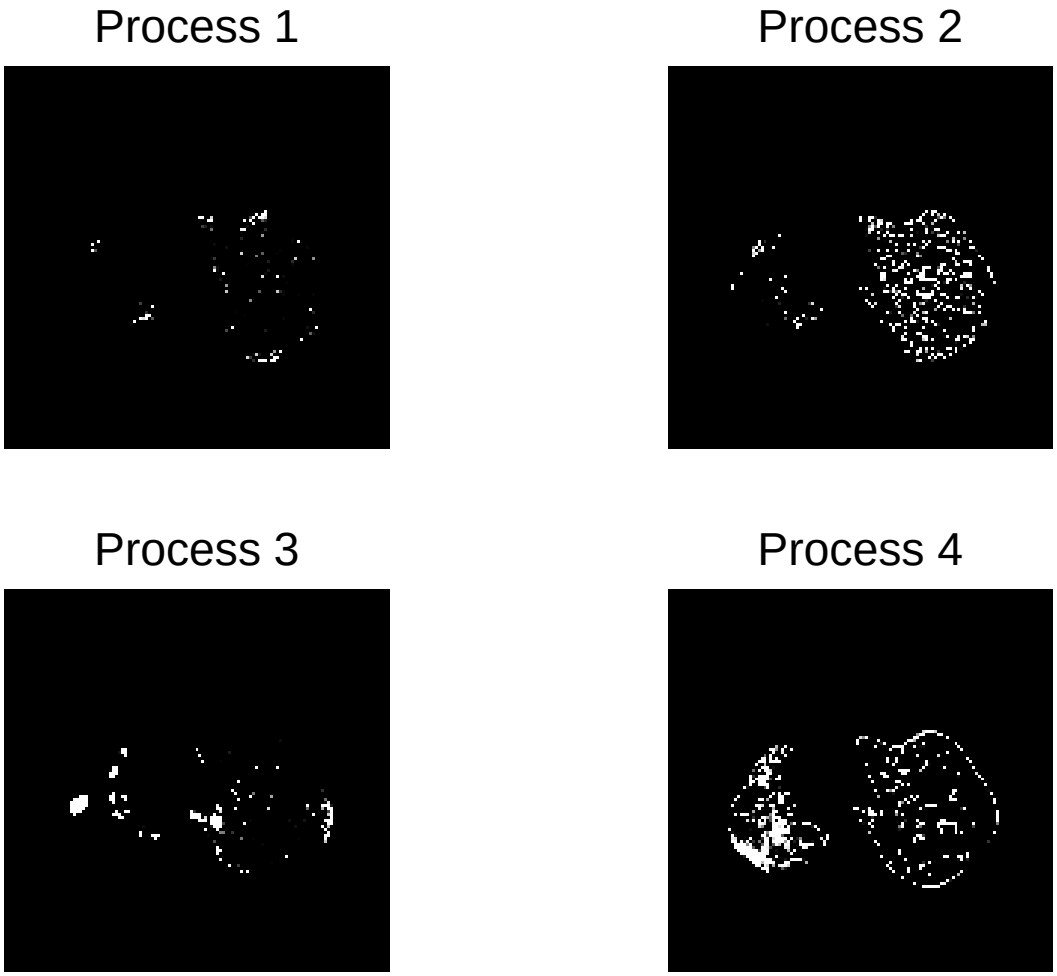


Fig. 4.6: Probabilities for membership to processes [1-4] for figure (4.5) in the 4x4 *Corr-LDA* model. Shown as a grey scale with white $\leftrightarrow P = 1$ and black $\leftrightarrow P = 0$.

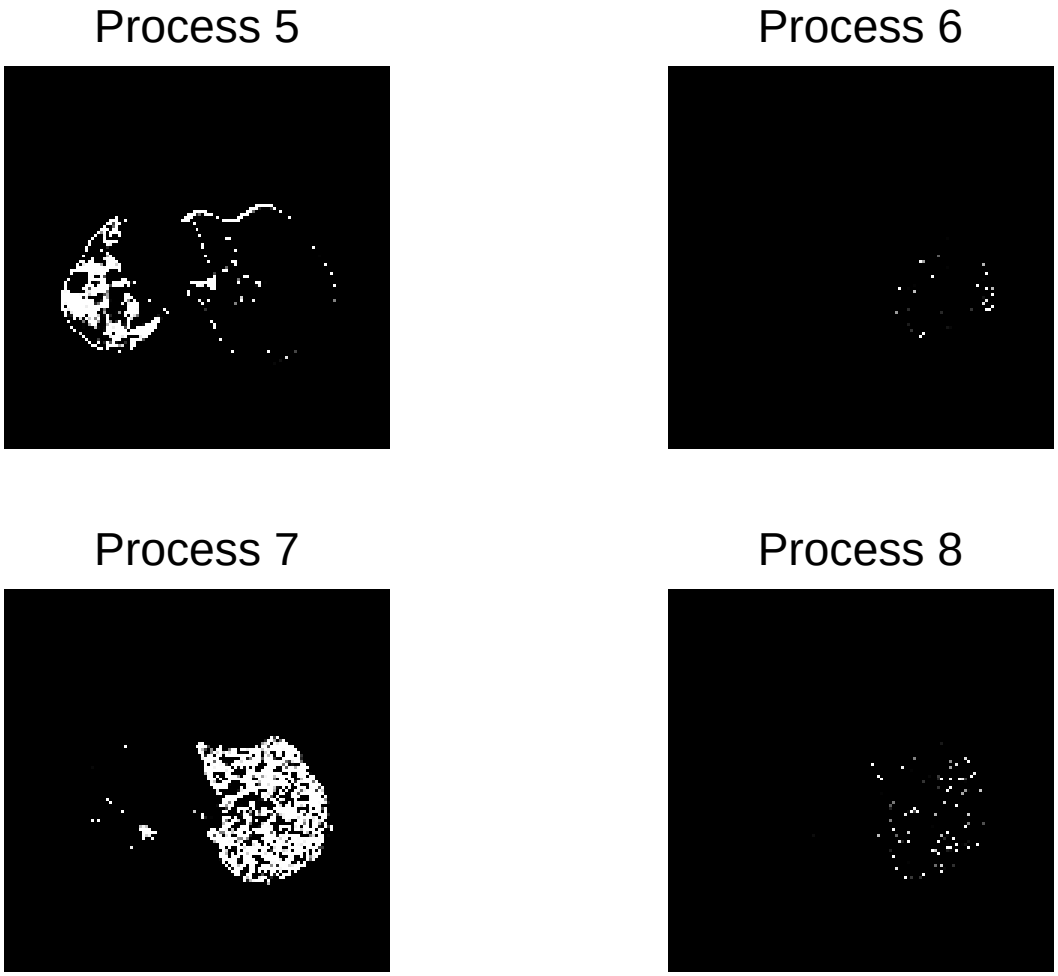


Fig. 4.7: Probabilities for membership to processes [5-8] for figure (4.5) in the 4x4 *Corr-LDA* model. Shown as a grey scale with white $\leftrightarrow P = 1$ and black $\leftrightarrow P = 0$.

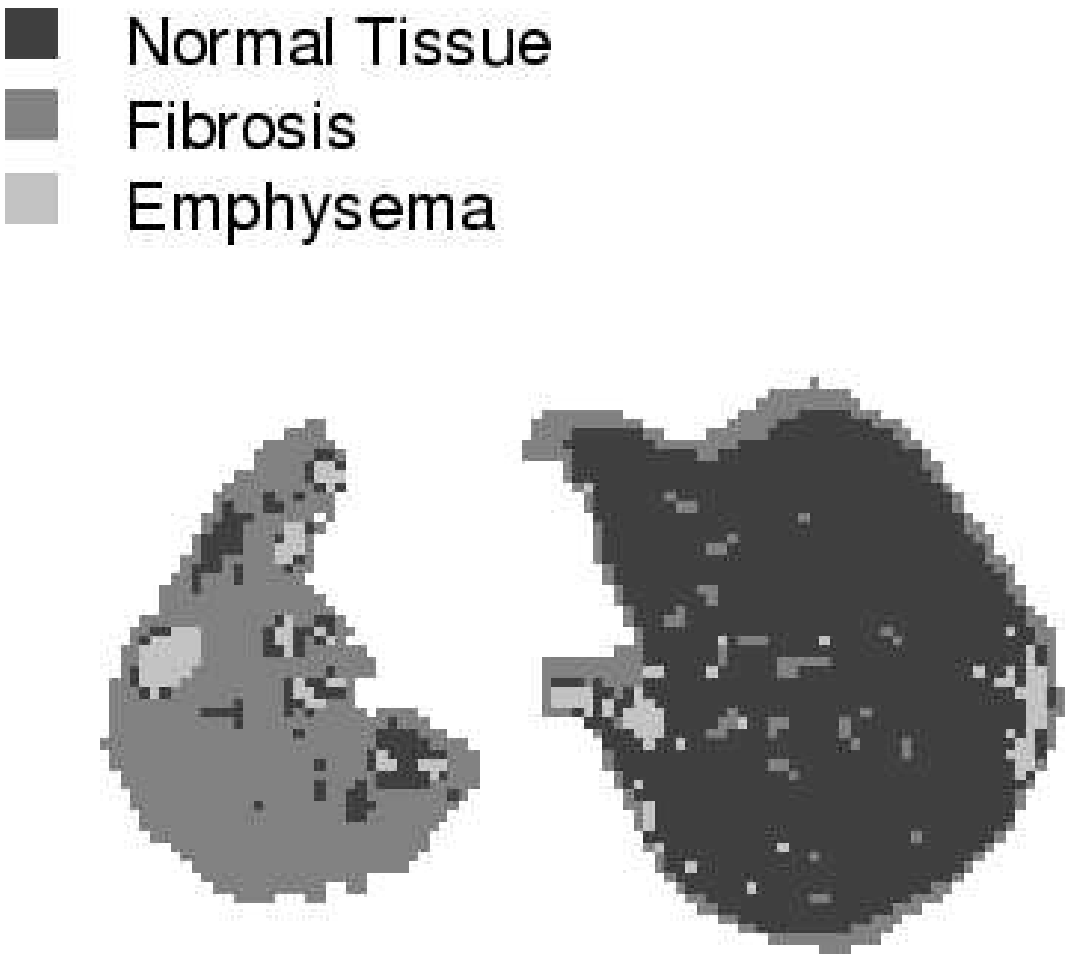


Fig. 4.8: SVM Classification of figure (4.5) for three classes using 4×4 regions sizes

Chapter 5

A Correspondence Model for the Joint Estimation of Motif and Gene Expression Data

5.1 Abstract

In this chapter we propose a generative probabilistic approach for the joint modelling of two types of data: gene expression values from microarray experiments and motif data describing binding site sequences in the upstream regulatory regions of genes. We compare four different strategies for this purpose, evaluating the performance of these algorithms on a microarray dataset for Saccharomyces Cerevisiae. We find correspondence models based on Latent Dirichlet Allocation are more appropriate representations to model the probabilistic relationship between motif abundance and gene expression levels.

5.2 Introduction

Currently many types of data are being generated which give different insights into the various functions of a genome. This includes continuous numerical data from microarray

experiments, discrete numerical sequence data and graphical information about regulatory networks. Models which successfully integrate these disparate data sources would be expected to give more insight into the underlying science than models which only utilise one type of data. This has motivated the development of new data fusion techniques.For example, Lanckriet *et al* [52] have successfully used kernel-based methods for this purpose. Microarray data can be handled using standard functional kernels (e.g. a linear kernel), sequence strings can be handled using string kernels and network (graph) information can be handled using a diffusion kernel [74]. Different types of data can therefore be incorporated into the model and prediction can be achieved using semi-definite programming to optimise the model parameters [52].

Rather than kernel-based methods, an alternative is to use generative probabilistic models. That is, models which could be used to produce the data. In this paper we will show that correspondence models of this type [15, 16] are superior at jointly modelling microarray gene expression data and motif data representing regulatory subsequences in the promoter regions of genes. We will show that such models can lead to prediction of gene expression levels from motif data or determine the relevance of particular motifs from a set of microarray gene expression ratios.

The objective of this work is to construct a model for the joint estimation of motif and gene expression data. The approach we discuss can also be extended to include numerical data beyond motifs, for example, expression values from regulator genes. In line with previous models proposed by the authors [70], we will use the word *process* to denote a set of assumed functionally related samples or genes.

In section 5.3 we introduce the data set. In section 5.4 we introduce the models and explain the assumptions behind these. In section 5.5 we describe two Correspondence models and in section 5.6 we will report numerical results. In section 5.7 we discuss these results and the extended framework enabled by this approach.

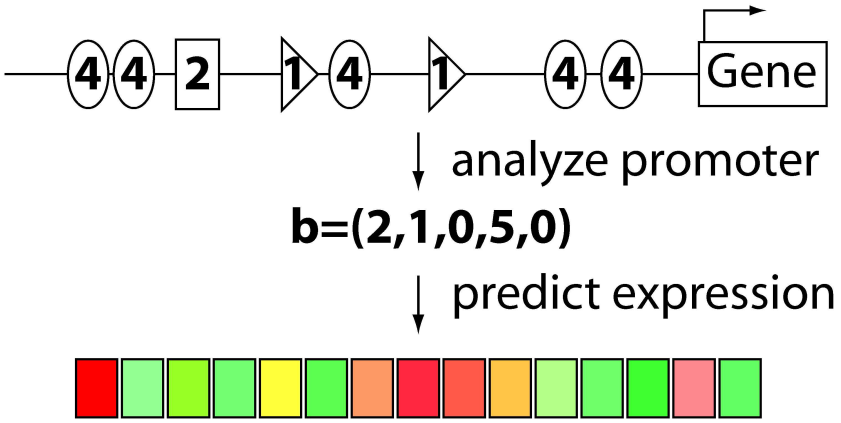


Fig. 5.1: Diagrammatic representation of the estimation of gene expression from motif data.

5.3 The Data Used

The data set we shall use was originally published in [36]. This a collection of expression values for the yeast Saccharomyces Cerevisiae across 173 experimental conditions. A summary of these experiments is given in table 5.1.

A single sample in the data set corresponds to a single gene and is made up of a pairing of

- 173 microarray expression values across the experiments listed in table 5.1.
- X motif counts.

The yeast cells are subject to 15 different classes of experimental condition. Joint modelling of the expression and motif structure

Exp IDs	Experimental Condition
1-29	Heat shock stress
30-35	Combined heat shock and osmolarity stress
36-45	Hydrogen Peroxide Stress
46-54	Superoxide generating drug Menadione Stress
55-69	Dithiothreitol (Disulfide-reducing agent) stress
70-77	Diamide (Sulfhydryl-oxidizing agent) stress
78-90	Hyperosmotic and hypoosmotic shock
91-95	Amino acid starvation stress
96-105	Nitrogen depletion stress
106-112	Diauxic Shift
113-134	Progression into stationary phase
135-139	MSN2/MSN4 and YAP1 deletion mutants with heat shock stress
140-144	MSN2/MSN4 and YAP1 deletion mutants with peroxide stress
145-147	MSN2/MSN4 and YAP1 over expression mutants
148-160	Steady state growth on alternative carbon sources stress
161-173	Steady state growth at constant temperatures

Tab. 5.1: Summary of the experimental details

5.4 The Models Used

The models we will consider are all closely related to mixture models [59].

We will assume that gene expression values are distributed under an experiment specific Gaussian distribution. For the motif counts we will form two classes of model: we assume either (a) a motif specific multinomial distribution or (b) a motif specific Poisson distribution. For (a) the motifs derive from a string with probabilities determined by a multinomial. As a trivial example, suppose we had five possible motifs for gene g, labelled 1 to 5, and the following string $\mathbf{a} = (4, 4, 2, 1, 4, 1, 4, 4)$ gives the occurrence of each in the upstream sequence, then the associated probability is $\prod_{m=1}^{8} P(M_{mg} = a_m)$ where $P(M_{mg} = a_m)$ is a multinomial and M_{mg} is the index of the mth motif in gene g. In this case the overall counts for each motif would be $\mathbf{b} = (2, 1, 0, 5, 0)$. For choice (b) we use the overall counts and the overall probability is $\prod_{n=1}^{5} P(C_{ng} = b_n)$ where $P(C_{ng} = b_n)$ is a Poisson distribution. Under the multinomial model we would only take account of motifs which are present but with the latter model we also take into account motifs which are absent (0 in $\mathbf{b}$), sometimes refereed to in probability theory as the *null event*. Under the normal distribution, given expression E_{dg} for gene g over experiment d we have:

$$P(E_{dg}|\Theta) \sim \mathcal{N}(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(E_{dg}-\mu)^2}{2\sigma^2}\right) \tag{5.1}$$

where Θ represents the set of distribution parameters, μ the mean and σ^2 the variance. The Poisson and multinomial distributions are then:

$$P(C_{ng}|\Theta) \sim Poisson(\beta) = \frac{\exp(-\beta)\,\beta^{C_{ng}}}{C_{ng}!} \tag{5.2}$$

$$P(M_{mg}|\Theta) \sim Multi(\eta) = \eta_m \tag{5.3}$$

where $C_{ng} = \sum_m \delta(M_{mg} = n)$, β is the Poisson mean and η_m is the multinomial parameter. These two assumptions for the motif counts will lead to different models. Also we will arrive at different models depending on our additional assumptions about the relationship between motifs and gene expression values. For an individual sample these assumptions will be about how expression values are related to each other, how motif counts are related and how the motifs and expression values are both related.

A *process* is defined by the set of model parameters indexed by a specific process number k μ_{gk}, σ^2_{gk} and β_{nk} or ν_{mk} (depending on the model assumptions).

In this paper we will consider four principal models (which we identify as **CorrM2E**, **CorrE2M**, **JMM** and **JMM-LDA** for our subsequent discussion). For the first (**CorrM2E**) gene expression is assumed generated by the existence of certain motifs in the promoter region according to our standard understanding of the biology. However, there is usually more information in the set of gene expression measurements than the set of motifs, so we also consider the inverse (**CorrE2M**) in which gene expression data is used to indicate the relevance of particular motifs. We will also consider two further models (**JMM** and **JMM-LDA**) in which both expression and motif are assumed generated from common underlying processes thus with an implicit connection of motif to expression rather than a direct relation.

In summary:

- **CorrM2E (Correspondence-Model: Motif to Expression)**: each motif is, in turn, generated by picking a process. The expression values are then generated by selecting a processes from the ones originally used to generate the motifs. Thus expression generation is conditioned on the motif processes.

- **CorrE2M (Correspondence-Model: Expressions to Motifs)**: each expression value is, in turn, generated by picking a process. The motifs are then generated by selecting a processes from the ones originally used to generate the expression values. Thus motif generation is conditioned on the expression processes.

- **JMM (Joint Mixture Model)**: the motifs and expression values are conditioned on a single process.

- **JMM-LDA (Joint LDA Mixture Model)**: each motif and each expression value can potentially derive from any process. The motif and expression processes are not explicitly linked and so could conceivably be conditioned on different processes.

The simplest of these models is a joint mixture model (**JMM**). The likelihood of a single sample, corresponding to gene g, with expressions $\mathbf{E_g}$ and motif counts $\mathbf{C_g}$, is given in equation 5.4.

$$P(\mathbf{E_g}, \mathbf{C_g}|\mathbf{\Theta}) = \sum_k \alpha_k P(\mathbf{E_g}, \mathbf{C_g}|\Theta_k) \tag{5.4}$$

where Θ represents the model parameters. By writing $\mathcal{G}$ as the whole data set, and using the standard variational approach shown in section 2.3.2 the overall log-likelihood can be lower bounded. This is given in equation 5.5:

$$\begin{aligned}\log P(\mathcal{G}|\mathbf{\Theta}) &= \textstyle\sum_g \log(\sum_k \alpha_k P(\mathbf{E_g}, \mathbf{C_g}|\Theta_k)) \\ &= \textstyle\sum_g \log(\sum_k \alpha_k \prod_d P(E_{dg}|\Theta_k) \prod_n P(C_{ng}|\Theta_k)) \\ &\geq \textstyle\sum_{gk} \gamma_{gk} \log\left(\alpha_k \prod_d P(E_{dg}|\Theta_k) \prod_n P(C_{ng}|\Theta_k)/\gamma_{gk}\right)\end{aligned} \tag{5.5}$$

Where $\sum_{k=1}^{n} \gamma_k = 1$ and γ_k is a *latent* variable. This lower bound is then maximised over all latent and model variables using an iterative EM-type algorithm. For **JMM-LDA** the likelihood is:

$$P(\mathcal{G}|\mu, \sigma, \beta, \alpha) = \int P(\mathbf{C}_g|\beta, \theta) P(\mathbf{E}_g|\mu, \sigma, \theta) P(\theta|\alpha) d\theta \tag{5.6}$$

As $P(\mathbf{C}_g|\beta, \theta)$ and $P(\mathbf{E}_g|\mu, \sigma, \theta)$ can be decoupled under a log, a similar approach to that given in [70] or [15] can be used to generate a lower bound on 5.6. Once again this is then maximised over all latent and model variables using an iterative EM-type algorithm.

5.5 A Correspondence Model

The first correspondence model we describe (**CorrM2E**) is based on the correspondence LDA model of Blei *et al* [15]. In this model there is sequential generation of the data. First the discrete motif data is generated and then the continuous gene expression data. Under the Poisson assumption in (5.2) the model can be summarised as follows:

1 Sample $\theta \sim Dirichlet(\theta|\alpha)$

2 For each motif n:

(a) sample a process $z_n \sim Multi(\Theta)$

(b) sample $C_{ng} \sim P(C_{ng}|z_n, \beta)$ conditioned on process z_n (using (5.2))

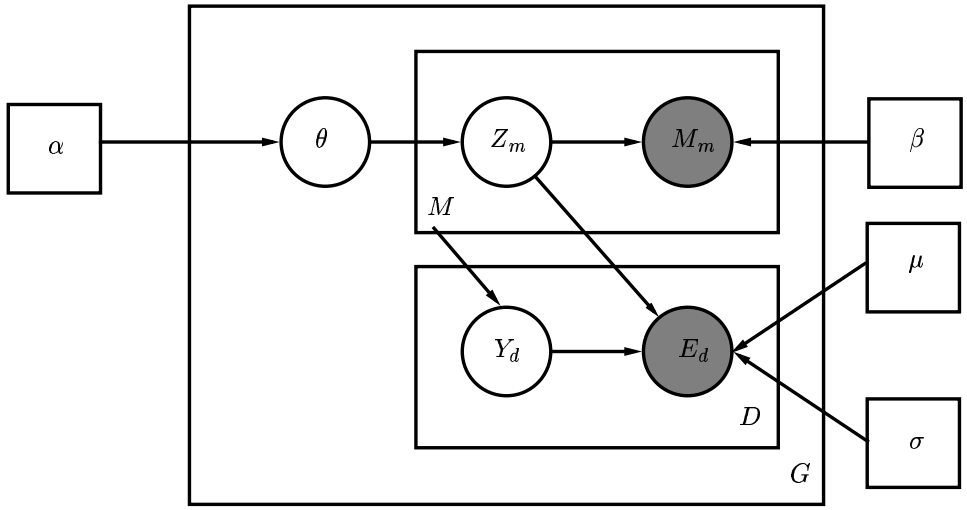


Fig. 5.2: A graphical representation of the generative correspondence model **CorrM2E**. We are performing a Maximum Likelihood estimate of the model parameters and so all such variables are represented by a square node.

3 For each experiment d:

(a) sample $Y_d \sim Uniform(1, \ldots, nomotifs)$

(b) sample $E_{dg} \sim P(E_{dg}|Y_d, \mathbf{z}, \mu, \sigma)$ conditioned on process $\mathbf{z}_{Y_d}$.

where *nomotifs* is the number of motifs. This sampling process is summarised in Figure 5.2.

For the multinomial assumption given in equation (5.3) this procedure is the same except C_{ng} is replaced by M_{mg} and n by m throughout. For **CorrE2M** the roles of motif and expression are reversed, with the assumption then being that the defining part of the data are the gene expression patterns. We will describe this model further in section 5.6.3. For **CorrM2E** and using the Poisson assumption given in (5.2), the likelihood of $\mathcal{G}$ as:

$$\begin{aligned} P(\mathcal{G}|\mu, \sigma, \beta, \alpha) &= \int P(\mathbf{E}_g, \mathbf{C}_g|\mu, \sigma, \beta, \theta) P(\theta|\alpha) d\theta \\ &= \int P(\mathbf{C}_g|\beta, \theta) P(\mathbf{E}_g|\mu, \sigma, \theta, \mathbf{C}_g) P(\theta|\alpha) d\theta \end{aligned} \tag{5.7}$$

where, letting $\theta_k = P(Z_n = k)$:

$$\begin{aligned} P(\mathbf{C}_g|\beta,\theta) &= \textstyle\prod_n P(C_{ng}|\beta,\theta) \\ &= \textstyle\prod_n \sum_k \theta_k P(C_{ng}|\beta, Z_n = k) \end{aligned} \tag{5.8}$$

and:

$$\begin{aligned} P(\mathbf{E}_g|\mu,\sigma,\theta,\mathbf{C}_g) &= \textstyle\prod_d P(E_{gd}|\mu,\sigma,\theta,\mathbf{C}_g) \\ &= \textstyle\prod_d \sum_{k,n} P(E_{gd}|\mu,\sigma,Y_d = n, Z_n = k) \\ &\times P(Y_d = n|\mathbf{C}_g)P(Z_n = k) \end{aligned} \tag{5.9}$$

A tractable lower bound on the likelihood in equation (5.7) can be given by using Jensen's inequality three times in succession. This introduces three latent variables γ_{gk}, Q_{ngk} and R_{dng}. These variables have the following interpretation. γ_{gk} is a k-dimensional gene specific Dirichlet parameter (normalised γ_{gk} gives the expected fraction of selections giving process k). Q_{ngk} is the probability that the nth motif count of gene g was generated by process k, and R_{dng} is the probability that the dth expression (from experiment d) of gene g was generated after selecting the nth motif. The lower bound is iteratively maximised using an EM-type algorithm based on the following set of update equations:

$$
\begin{aligned}
\mu_{dk} = &\quad \frac{\sum_{g,n} Q_{ngk} R_{dng} E_{dg}}{\sum_{g,n} Q_{ngk} R_{dng}} \\
\sigma^2_{dk} = &\quad \frac{\sum_{g,n} Q_{ngk} R_{dng} (E_{dg} - \mu_{dk})^2}{\sum_{g,n} Q_{ngk} R_{dng}} \\
\beta_{nk} = &\quad \frac{\sum_g Q_{ngk} C_{ng}}{\sum_g Q_{ngk}} \\
R_{dng} \propto &\quad \exp\left[\sum_k Q_{ngk} \log P(E_{dg} | Y_d = n, Z_n = k, \mu, \sigma)\right] \\
&\quad \times P(Y_d = n | \mathbf{C_g}) \\
Q_{ngk} \propto &\quad \exp\left[\sum_d R_{dng} \log P(E_{dg} | Y_d = n, Z_n = k, \mu, \sigma)\right] \\
&\quad \times P(\mathbf{C}_g | Z_n = k, \beta_{nk}) \exp\left[\Psi(\gamma_{gk}) - \Psi(\sum_k \gamma_{gk})\right] \\
\gamma_{gk} = &\quad \alpha_k + \sum_n Q_{ngk} \\
\alpha_{new} = &\quad \alpha - H(\alpha)^{-1} g(\alpha).
\end{aligned}
\tag{5.10}
$$

where $H(\alpha)$ is the Hessian and $g(\alpha)$ is the gradient, calculated as derivatives of the likelihood expression (see [15] or [70] for details). The special form of the Hessian means the inverse Hessian does not need to be evaluated explicitly (see Appendix of Blei *et al* [15] for details).

So far we have discussed a maximum likelihood (see section 2.1.6) approach with a uniform prior implicitly assumed. However, we can adopt a non-uniform prior on the model parameters (see [70] for examples). This will act as a smoother and go some way to avoiding over-fitting. A fully Bayesian approach would be to treat the model parameters as random variables, rather than point estimates. A variational EM would not be sufficient in this case a MCMC or variational Bayes approach to parameter estimation would be needed. In this case MCMC would be computationally infeasible.

A number of choices are possible but a reasonable priors on μ and σ^2 would be $P(\mu) \propto N(0, 1)$ and an Inverse Gamma probability distribution respectively.

5.6 Experimental Results

For our experiments we will use the dataset of Gasch *et al* [36]. The gene expression measurements (which are $\log_2$ of the expression ratios) were taken over a total of 1411 genes across 173 experiments. They derive from cDNA microarray experiments recording response to environmental stresses for the organism *S. Cerevisiae*. This dataset has been supplemented with 354 binding site motifs by Middendorf *et al* [61] using 500 bp sequences drawn from the Saccharomyces Genome Database (SGD) and filtered for motifs using the PATCH tool in the TRANSFAC database [88]. From these we selected 200 motifs by discarding those which had less than 10 occurrences in the data. Compared to Middendorf *et al* [61] we use continuous rather than discretized expression data.

5.6.1 Model Comparison

First we need to determine which of the various models gives the best generalisation in a likelihood sense. In Figure 5.3 and 5.4 we have performed a 5-fold cross validation study of the the logged likelihood versus the number of processes. During the cross validation 20% of the data is retained during parameter estimation and then the average likelihood of the left-out data is calculated. This involves approximating the likelihood in equation (5.7) by replacing the integration by a summation with multiple sampling of θ. This cross validation study also indicates what a reasonable choice for the number of processes would be. We have used a maximum likelihood approach (a uniform prior, hence the model overfits the data after passing through a peak). In the experiments conducted by Gasch *et al* [36], 16 different types of environmental stress were used (give in table 5.1). However, some of these stress modes are likely to be similar in action. Consequently, the peak at 10 processes could well indicate the number of processes required to adequately model this set of stress responses.

These two figures indicate that the correspondence models outperform the mixture models. In the next section we will therefore focus on the correspondence models. Apart from a comparison between models we also compared performance against a null model and nearest neighbour models. For the null model we used the same data but we assume that only a single process is used. In this case, with the choice of a single process only, all the models

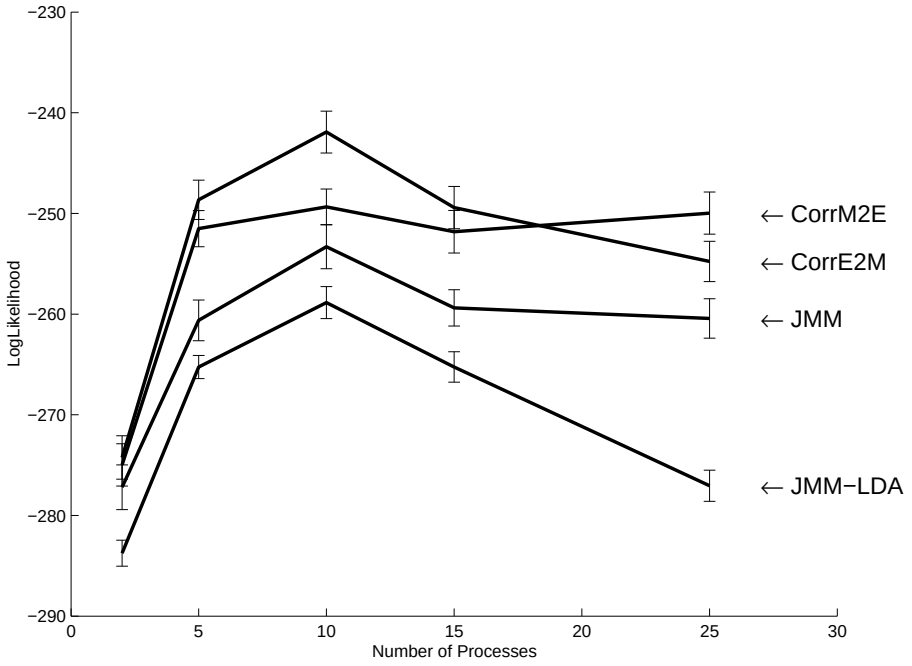


Fig. 5.3: Log Likelihood (y-axis) versus number of processes (x-axis) using a model based on the Poissonian distribution of the motif counts, equation (5.2).

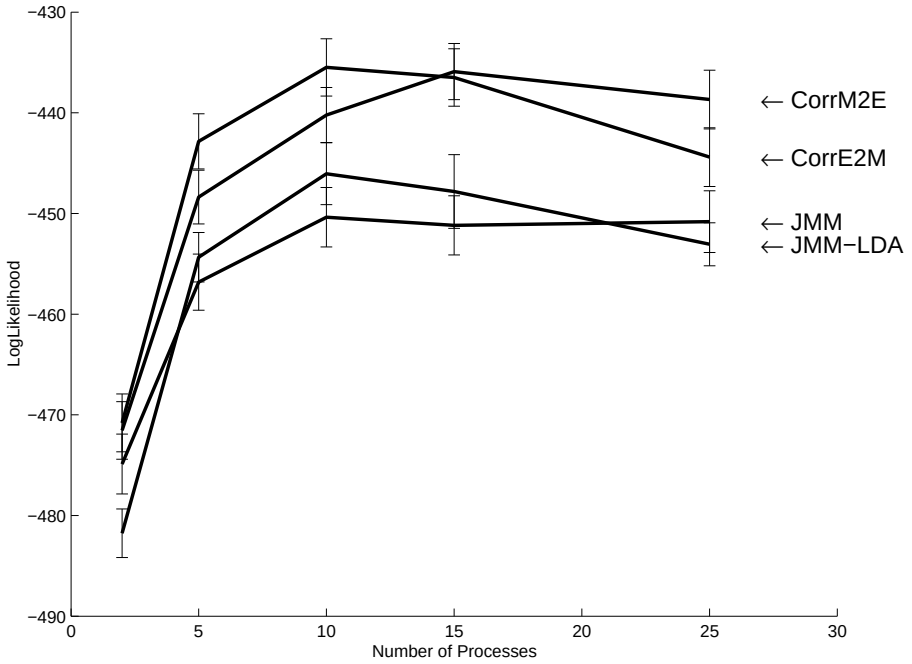


Fig. 5.4: Log Likelihood (y-axis) versus number of processes (x-axis) using a model based on the multinomial distribution for the motifs, equation (5.3).

CorrM2E, **CorrE2M**, **JMM** and **JMM-LDA** are identical. For **CorrM2E** the predicted expression value is the average expression value, taken across the samples, for the given gene. For **CorrM2E** the prediction is the average motif count, taken across the samples, for the given motif. This is a extremely naive model but it gives a reasonable minimum benchmark. Using 5-fold cross-validation the log-likelihood calculated as -496.07 ± 2.58 in Figure 5.4 with the multinomial assumption and -325.95 ± 2.75 in Figure 5.3 with a Poisson assumption. Relative to this benchmark the use of a Poisson model has a gain of over 40 points on the log-scale. The multinomial model has less of a gain but it is still notably better than using an averaged expression value. For this reason we will concentrate on use of a Poisson distribution in the following.

As an alternative, we also investigated two k nearest neighbour models, estimating the expression value of a gene using the closest k genes based on motif profile. For the first model, the k nearest neighbours were found using the Euclidean distance between the given gene's motif profile (a string of integers for motif counts) and the motif profiles of other genes. The second model was probabilistic. Thus, under a Poissonian model, these k nearest motif profiles are chosen such that, if their values are taken as the Poisson parameters, they would maximise the likelihood of the given gene's profile.

The likelihood curves for both these models are given in Figure 5.5. As we increase k the likelihood curves will tend to the value given in the null model discussed above as the null model is in a sense a nearest neighbour methods in which you consider all neighbours. Though using the profiles of a set of nearest neighbours could be expected to give a better model than a model using the profiles for all genes, such nearest neighbour models can be adversely affected by the occurrence of very different gene expression profiles associated with very similar motif profiles (illustrated later in Figure 5.14).

So far we have shown that the Correspondence models perform better than similar simpler models and naive approaches. This is only a relative measure of performance and does not indicate an absolute measure of success.

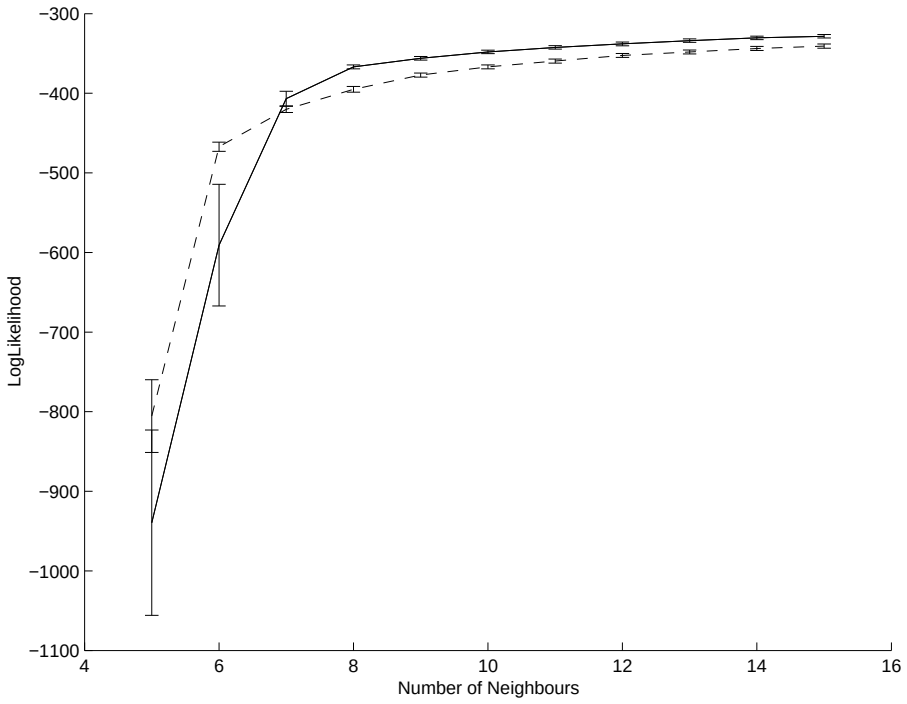


Fig. 5.5: Log Likelihood (y-axis) versus number of neighbours, k, (x-axis) for estimated expression values based on averaging of expression over the k nearest motif profiles. The solid curve is for the probabilistic model mentioned in the text and the dashed curve is for the non-probabilistic model based on use of a Euclidean distance to determine nearest neighbours.

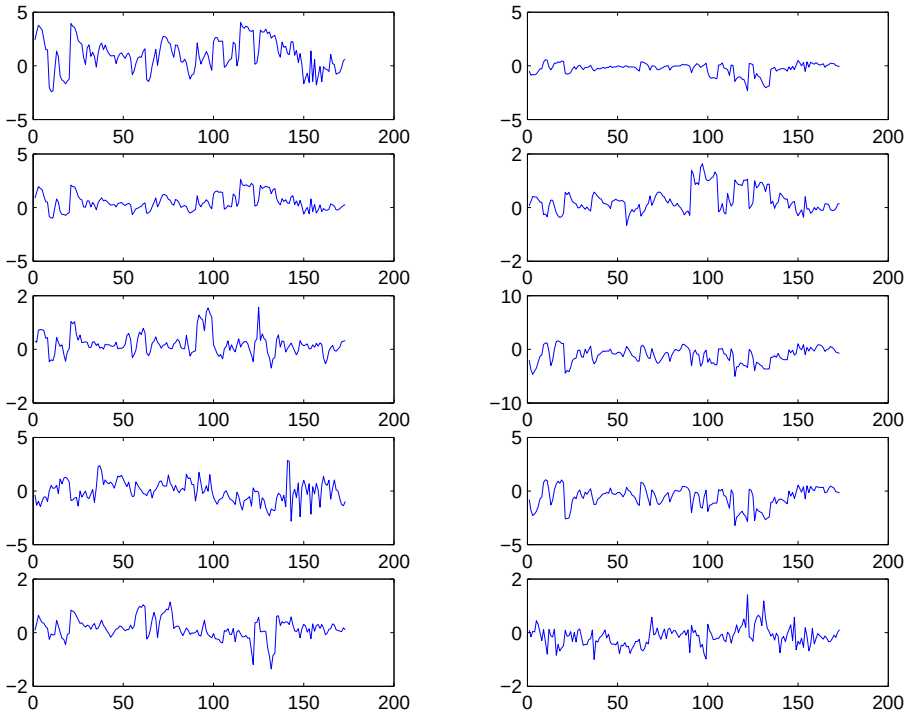


Fig. 5.6: The means μ_{dk} for the Motif to Expression correspondence model **CorrM2E**. The x-axis gives the $d = 1, \ldots, 173$ experiments for processes $k = 1, \ldots, 10$.

5.6.2 The Correspondence Model CorrM2E

To use the **CorrM2E** for prediction we left out 10% of the whole data set. Then using the variational EM update equations 5.10 we estimated the model parameters on the remaining 90%. Figure 5.6 gives the means μ_{dk} across the $d = 1, \ldots, 173$ experiments of Gasch *et al* [36], for all $k = 1, \ldots, 10$ processes. Similarly the poisson model parameter β is given in 5.7.

Using the estimated model parameters, μ, β, σ^2 and α we can take the motifs from left out 10% of data and try to predict the corresponding expression values. To do this prediction we need to compute the density

$$P(E_{gd}|\mu, \sigma, \theta, \mathbf{C}_g) = \sum_{k,n} P(E_{gd}|\mu, \sigma, Y_d = n, Z_n = k) P(Y_d = n|\mathbf{C}_g) P(Z_n = k|\alpha, C_{ng})$$

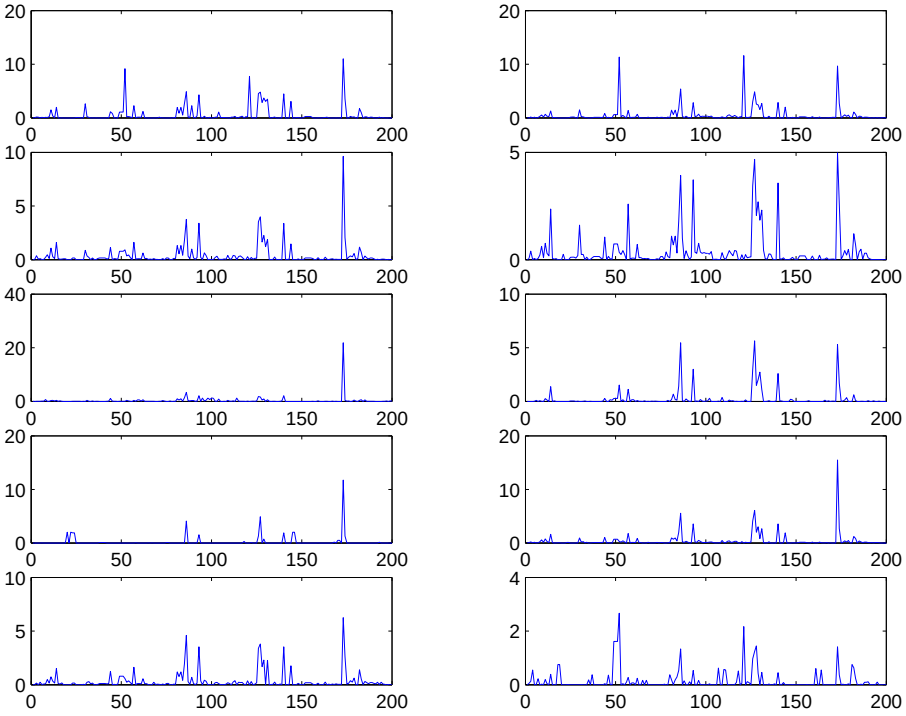


Fig. 5.7: The Poisson mean β_{mk} for the Motif to Expression correspondence model **CorrM2E**. The x-axis gives the $m = 1, \ldots, 200$ motifs for processes $k = 1, \ldots, 10$.

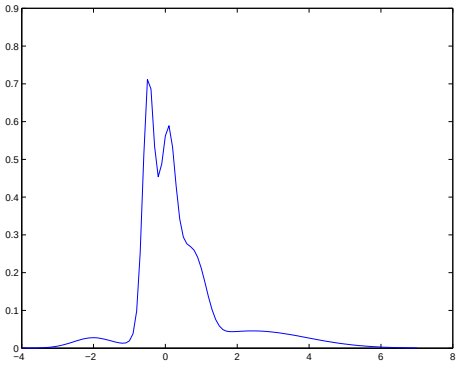


Fig. 5.8: Predicted density for $\mathbf{E}_g$ given the motifs.

originally given in equation 5.9. $P(Z_n = k|\alpha, C_{ng})$ is calculated by using Bayes Rule. The resulting density $P(E_{gd}|\dots)$ will be a mixture distribution like that given in figure 5.8, we shall therefore take the mode of this mixture as the prediction.

In figure 5.10 we give a scatter plot of the predicted expression values against the actual values for the held out 142 genes across each of the 173 experiments. A histogram of the corresponding correlation coefficients for predicted vs actual is given in figure 5.10. From these two plots it is clear that the model has very weak predictive powers. We shall now investigate why, despite having the highest cross validated likelihood the model gives poor results.

Investigation

Figure 5.11 is a plot of the normalised latent variable γ for each held out gene. Recall, γ_g is a sample specific Dirichlet parameter, and so on normalisation this will give us the expected number of times process k was selected in generating the motif counts for gene g. From figure 5.11 we see that there is substantial mixing between processes, and very rarely is any particular held out gene strongly associated with a single process. This indicates that there are only weak clusters within the motif data. In figure 5.12 we have plotted the relative Poisson means for each experiment across the 10 processes. This is to indicate the difference between the processes at the motif generation level. A number of processes seem to indicate differences but these will probably be drowned out by the dominant motifs (see the peaks of

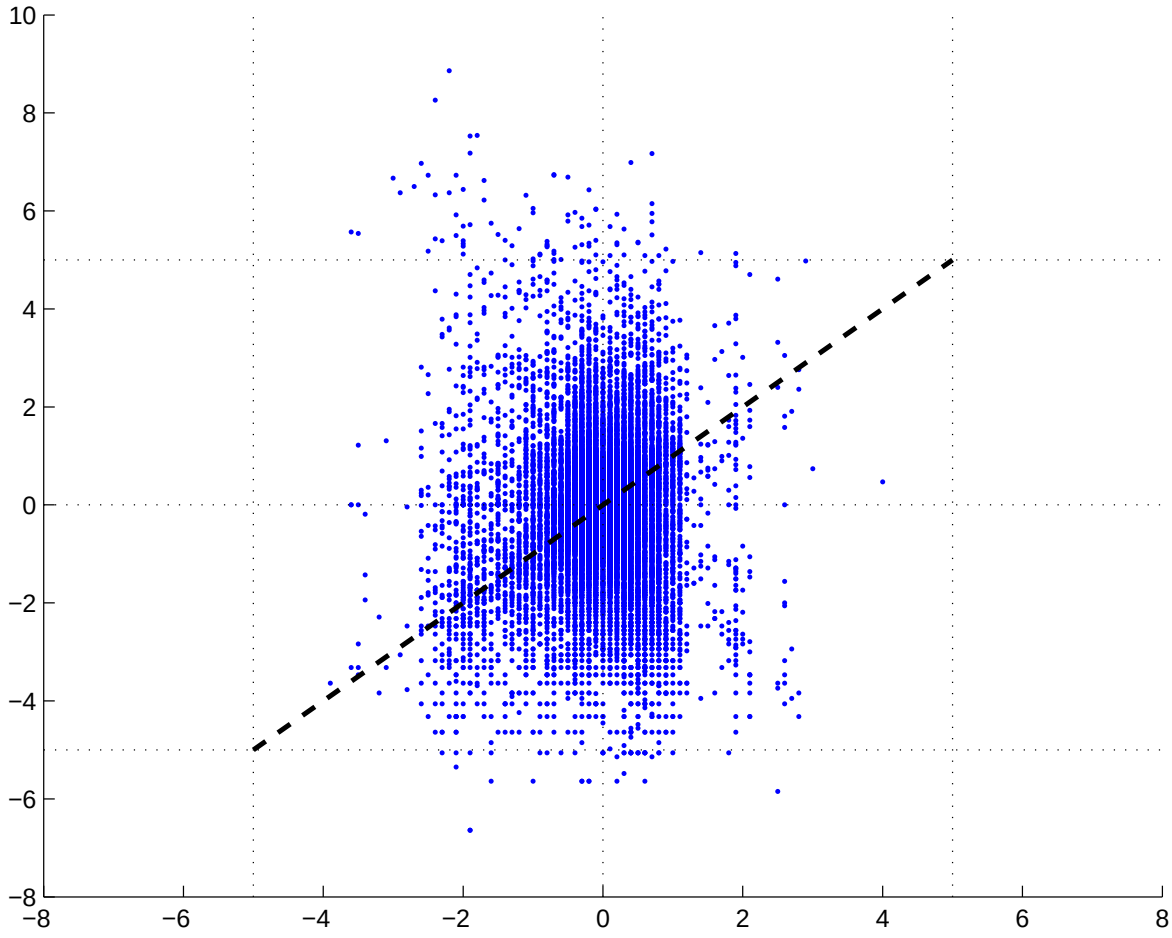


Fig. 5.9: Scatter plot giving the predicted value (x-axis) versus the actual value (y-axis) across 142 genes from 1411, with 173 experiments per gene.

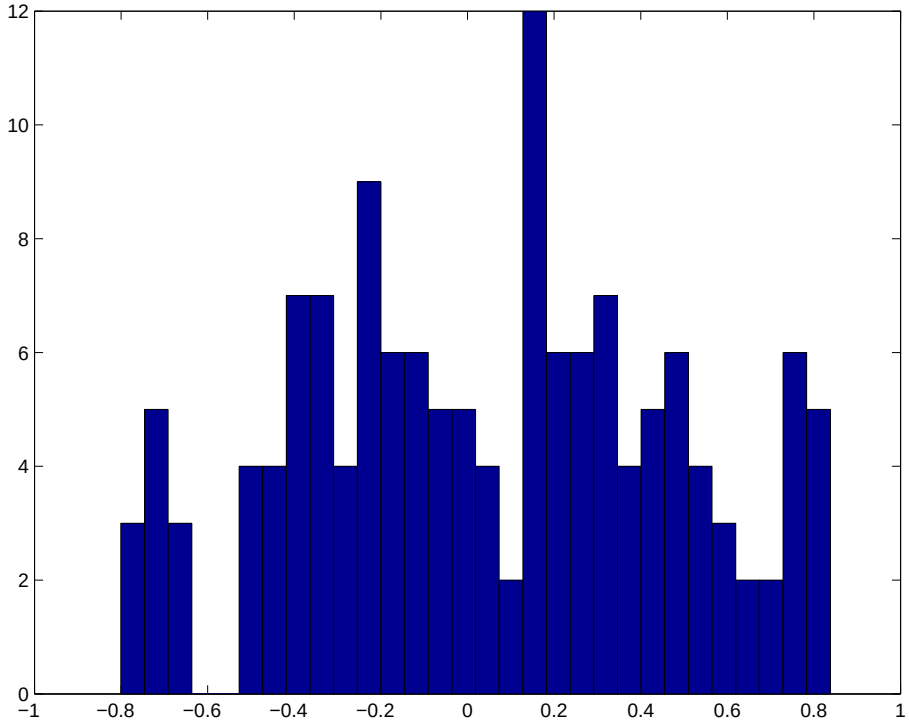


Fig. 5.10: A histogram giving the number of occurrences (y-axis) versus correlation coefficient (x-axis) for 142 randomly selected held-out genes from 1411. The correlation coefficient is between predicted and actual gene expression values.

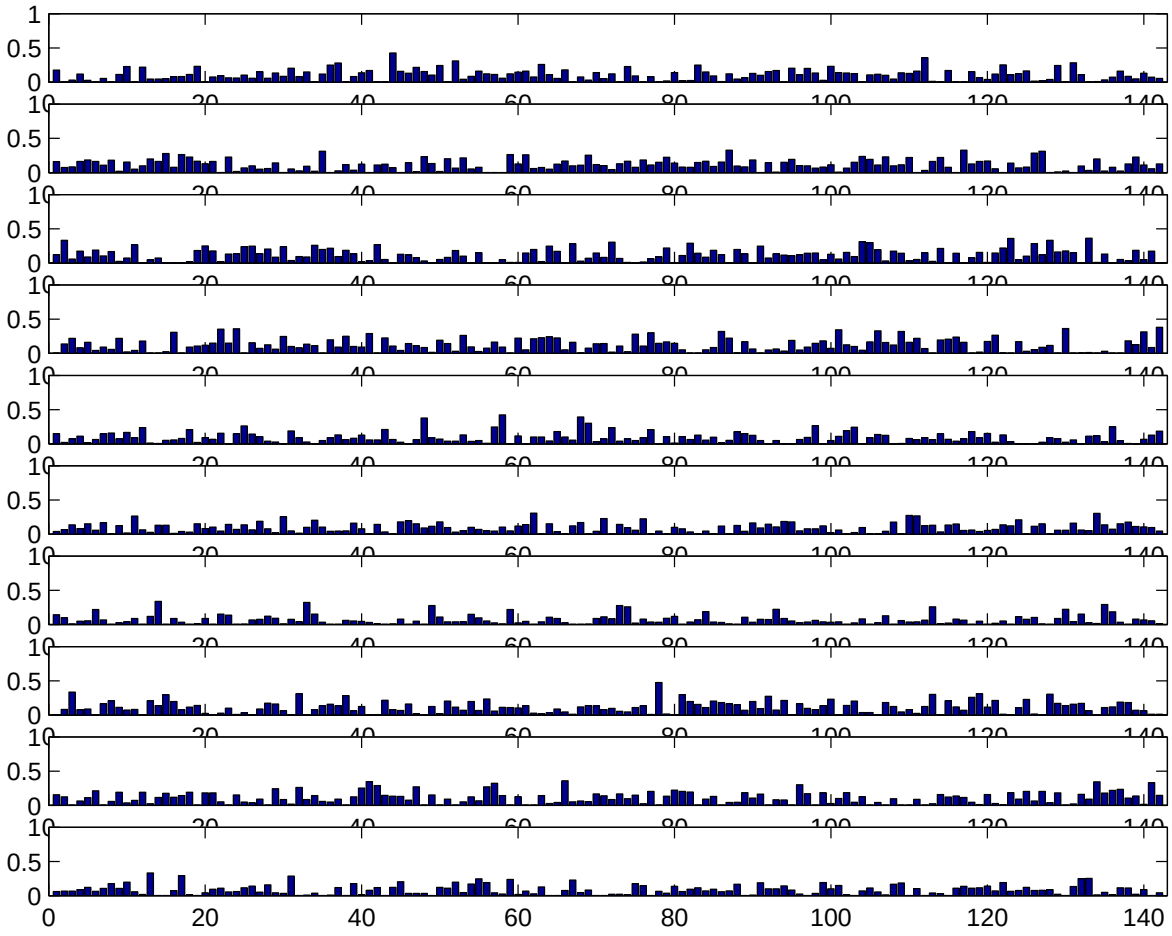


Fig. 5.11: Normalised bar-plot of the latent variable γ for 142 held out genes from 1411, with 173 experiments per gene.

figure 5.6 for dominant motifs).

There are a number of possible reasons for poor predictive performance, some of which we list below.

- The motif data is incomplete.
- Subtle changes in motifs which indicate expression are drowned out by noise from uninformative motifs.
- Motif data is unrelated to expression data.

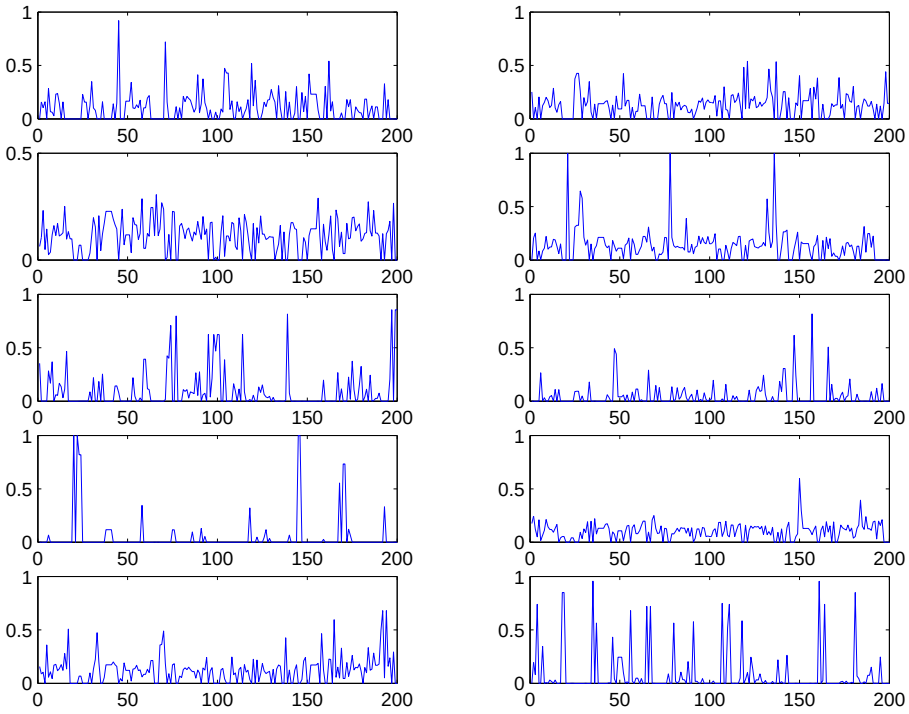


Fig. 5.12: The relative (normalised across processes) Poisson mean β_{mk} for the Motif to Expression correspondence model **CorrM2E**.

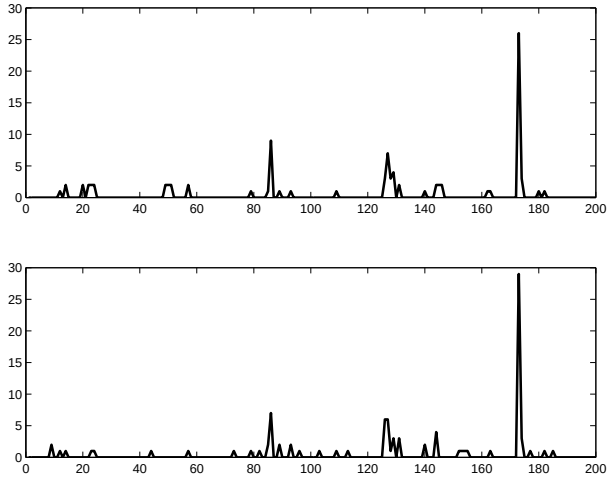


Fig. 5.13: Two examples showing two very similar motif profiles.

As we are limited by the original data set, and also computationally limited we shall ignore the first point. We shall also discount the last point as without that there is not basis for investigation. We shall concentrate on the second point. By writing $corr(E_a, E_b)$ and $corr(C_a, C_b)$ as the pairwise (gene a to gene b) correlation between expressions and motifs respectively, we searched through the full set of genes to find

$$max_{a,b}||corr(C_a, C_b) - corr(E_a, E_b)||$$

The plot of the motifs for each of these genes is given in figure 5.13, and the corresponding expression profile is given in figure 5.14. As can be seen the motif profiles are very very similar, and certainly within the framework of a probabilistic model (like the **CorrM2E**) would appear similar in a distributional sense. But they have virtually antisymmetric expression profiles.

In the publication of Middendorf *et al* [61] they achieve some level of success in held out prediction on the same data set. Their approach to prediction was to train alternating decision trees. A small section of the leaf nodes of one of their decision trees is given in figure 5.15. The node in the bottom left is splitting on the presence / absence of motif **MY$CYC1_16**, just this single motif makes the difference between an up regulation of 2.5

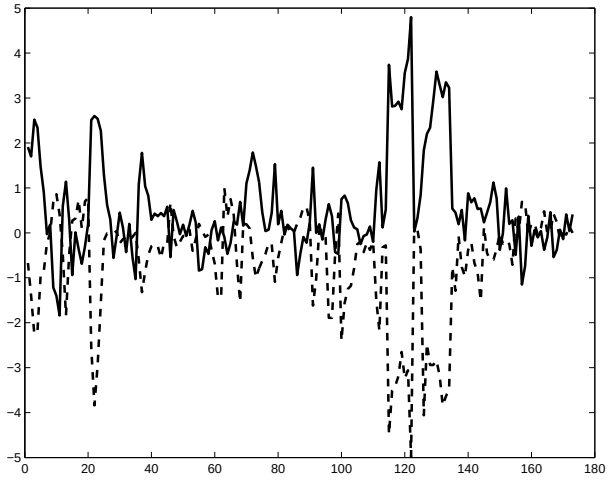


Fig. 5.14: Expression profiles for two examples. Subtle differences in the set of motifs in 5.13 can lead to very different expression profiles. In 5.13 the two sub figures show two very similar motif profiles. However, the derived expression profiles are very anti-correlated. Note that these profiles come directly from the data and are not derived from the algorithm.

or a down regulation of -1.3. In other words even a motif count of 1 for **MY$CYC1_16** will switch expression from substantially up to substantially down. The probabilistic models given above will always struggle to give accurate predictions if this is really the relationship between motifs and expressions. A proposed model would have to be more complex and have greater non linearity to be rich enough to model these details.

Mixture Fitting

In this section we shall not perform prediction, but we will see how well **CorrM2E** jointly models the data. We are in a sense finding the best combination of processes to simultaneously fit the 200 motif counts and 173 expression values for a given gene. This fitting can be done by, for each held out gene run the update equations (from equation 5.10) for the *latent* variables. That is *gamma*, Q and R. Figure 5.16 gives a normalised plot of the resulting γ. Already we can a see much stronger clustering than we saw for prediction (see figure 5.11). This shows that there exists far more structure in the expression profiles than in the motifs counts.

To illustrate the idea of the mixture fitting, in Figure 5.18 we give an example in which the

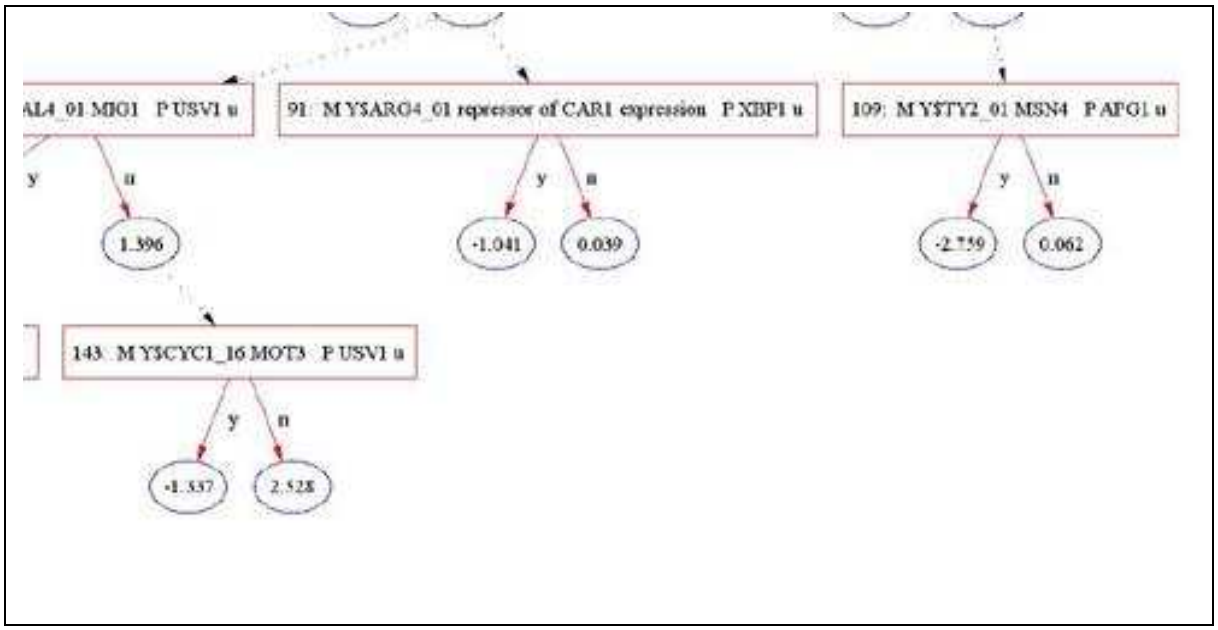


Fig. 5.15: An example subsection of the decision trees published in Middendorf *et al* [61].

probability of membership of 7 processes is effectively zero and the set of expression values for the given gene (SRM1) can be represented by three processes.

Thus for experiment d we obtain a mixture density with means μ_{dk} and standard deviations σ_{dk} taken over those k for which normalised γ_{gk} is non-zero. For experiment $d = 127$ in Figure 5.18 we illustrate this mixture density in Figure 5.19. The reason there exists a three component mixture, and rather than a single component which is closest to the actual value is two fold. Firstly we are simultaneously fitting a mixture to the motifs, which will add weight to some processes, and secondly the Dirichlet model parameter α will also influence which processes are selected.

In Figure 5.18 we note that the bottom process is a reasonable representation of expression (the solid and dashed curves are fairly aligned). The middle process appears to be a poor representation: in fact, the predicted and actual expression curves appear anti-correlated in the region of experiments 90-140. However, in Figure 5.19 we see that the middle process appears with an associated large variance. We use a mixture $\Phi = \sum_k p_k \mathcal{N}(\mu_{dk}, \sigma_{dk})$ where p_k represents the probability of process membership (normalised γ_{gk}) and the fitted expression value will taken as the mode, or maximum value achieved within this distribution.

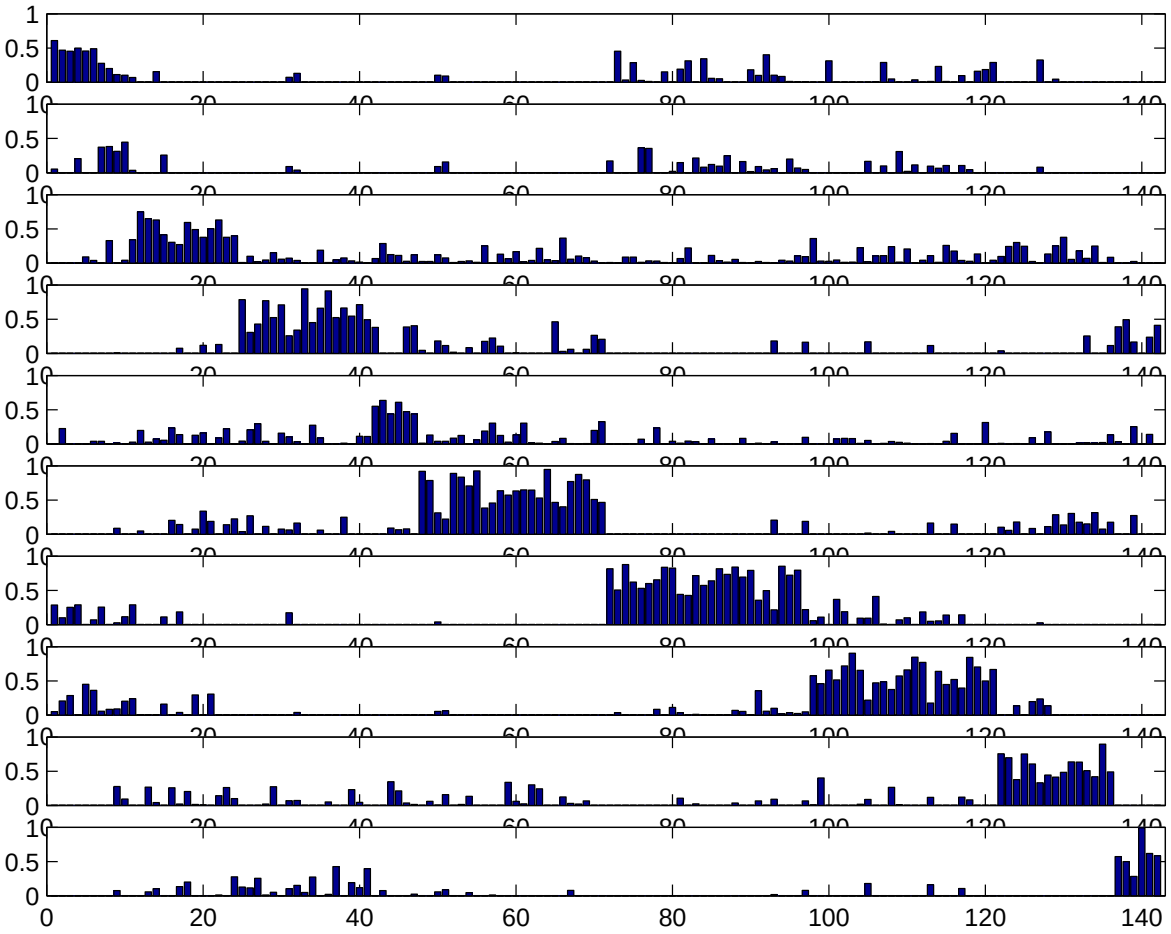


Fig. 5.16: Reordered normalised bar-plot of the latent variable γ for 142 held out genes from 1411.

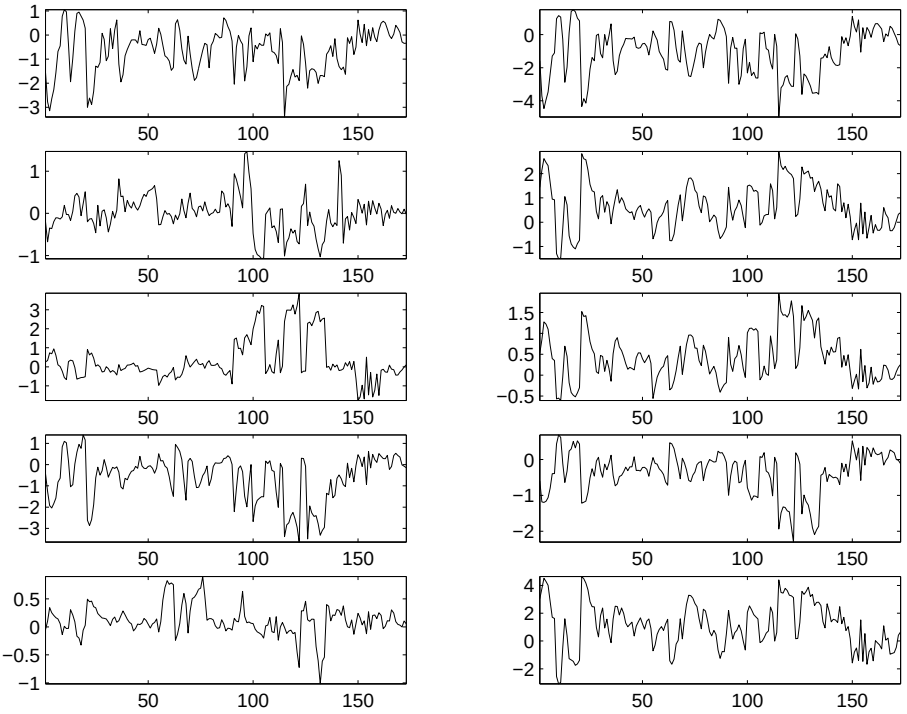


Fig. 5.17: The means μ_{dk} for the Motif to Expression correspondence model **CorrM2E**. The x-axis gives the $d = 1, \ldots, 173$ experiments for processes $k = 1, \ldots, 10$.

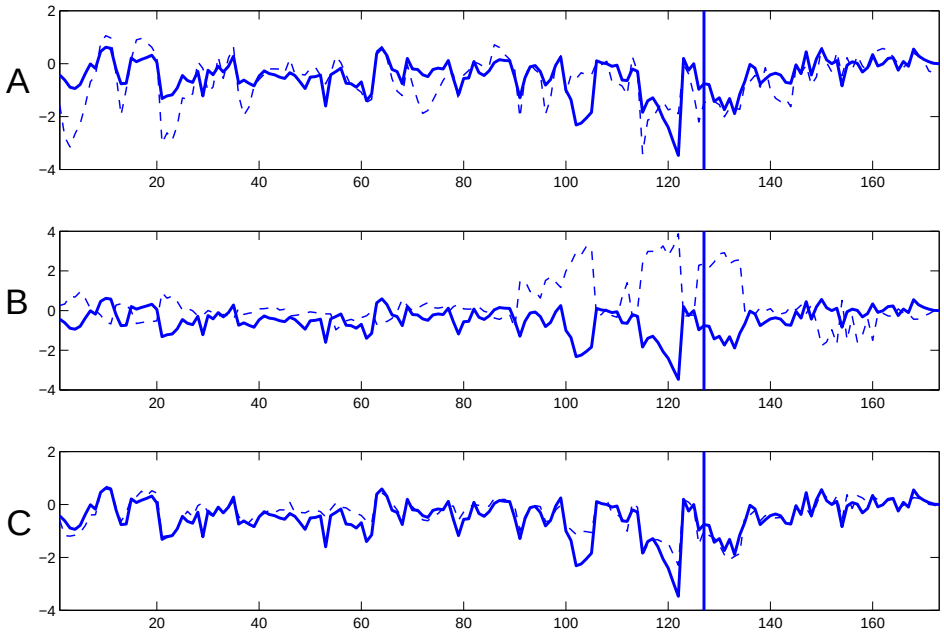


Fig. 5.18: In this case for each process the model samples with a probability (normalised γ_{gk}) of membership of 0.18 for the top process, 0.31 for the middle process and 0.49 for the bottom process. Along the x-axis we have the experiment number d. The solid curve gives the actual expression values for the hold-out gene (SRM1) and the dashed curve would be the fitted value *were expression represented by this process only.*

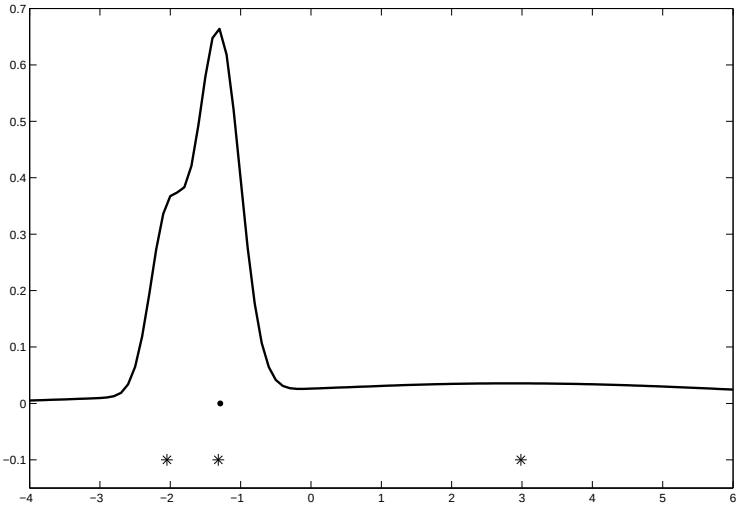


Fig. 5.19: Mixture density derived for experiment 127 in Figure 5.18. The curve derives from μ_{dk} and standard deviations σ_{dk} for the given experiment $d = 127$ and the three process k. The solid upper circle denotes the actual expression value and the lower three stars are the associated means for the top (left star), middle (right star) and bottom (middle star) process in Figure 5.18.

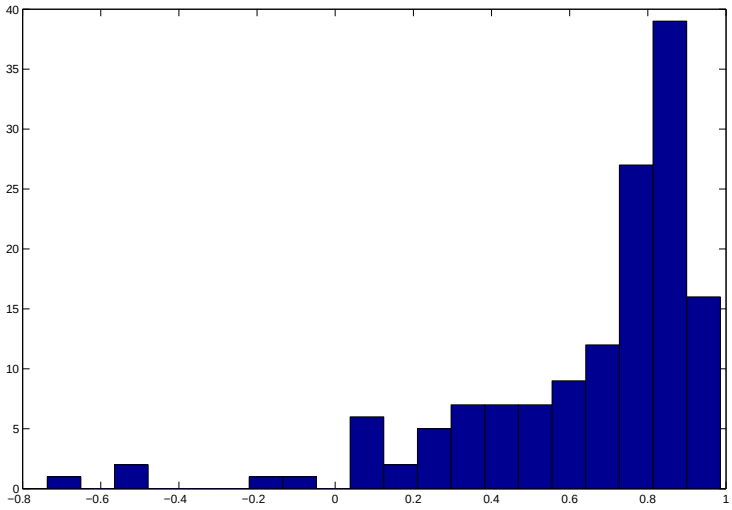


Fig. 5.20: A histogram giving the number of occurrences (y-axis) versus correlation coefficient (x-axis) for 142 randomly selected held-out genes from 1411. The correlation coefficient is between predicted and actual gene expression and the prevalence of correlation scores near 0.8 indicates reliable prediction.

Using the mode and fitting the expression value on 142 held-out genes from 1411 we get the distribution of correlation coefficients given in Figure 5.20. This is for comparison to figure 5.10 for the predicted case.

To visually indicate the extent of correlation between fitted and actual we give a scatter plot in Figure 5.21 across these 142 genes. In Figure 5.22 we give an illustration of fitted versus actual gene expression on three hold-out genes using **CorrM2E**.

In Figure 5.23 we give the β spectrum across the 10 processes and 200 motifs. Since β in (5.2) gives the mean and variance of the distribution, the curves indicate the average motif profile for that process. One motif is observed to be dominant in all processes.

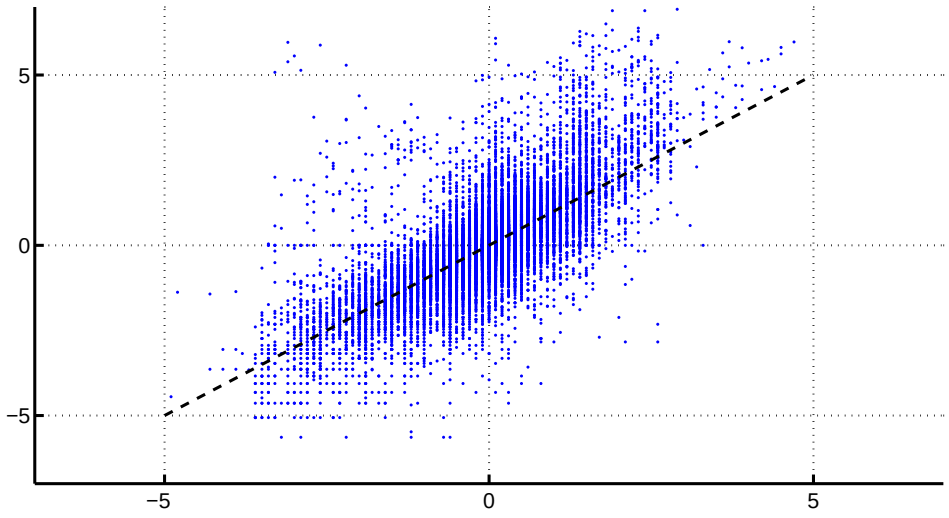


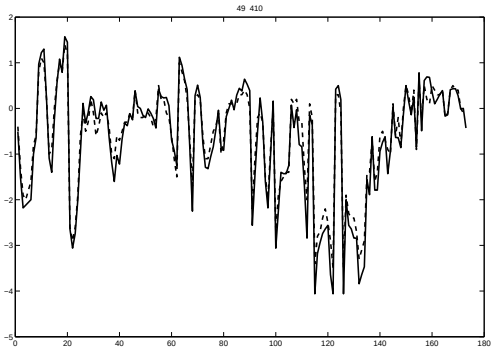
Fig. 5.21: Scatter plot giving the fitted value (x-axis) versus the actual value (y-axis) across 142 genes from 1411, with 173 experiments per gene.

5.6.3 The Correspondence Model CorrE2M

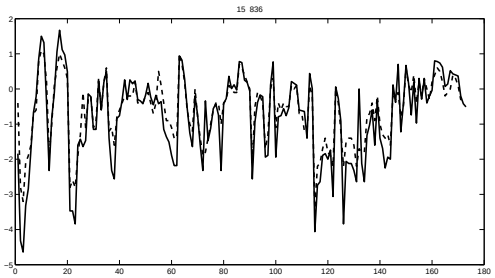
We can also use the Correspondence model to predict relevance of certain motifs given gene expression data. As an illustration, in Figure 5.24 we give an example in which the algorithm draws from three processes to model the actual motif profile given in the top subplot.

5.7 Conclusion

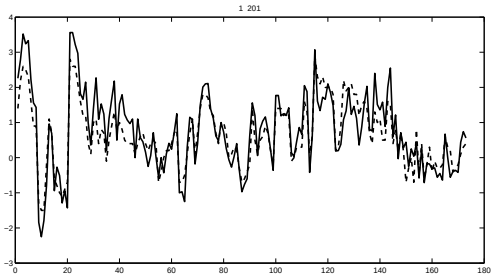
We have proposed a framework for the joint modelling of microarray gene expression and motif data. In terms of practical applications, accurate prediction of gene expression means from the motif structure does not seem easy within a probabilistic framework. Indeed the failure to predict expression could be used as a starting point for iterative refinement of a procedure to better capture the motif structure [71]. The comparisons in Figures 5.3 and 5.4 suggests that correspondence models work better than standard mixture models. Furthermore this correspondence framework can be readily extended to handle other types of data. For example, rather than motifs, we may have knowledge about which parent genes



(a) RPL7A



(b) NTH1



(c) GAD1

Fig. 5.22: Three examples of fitted (dashed curve) versus actual (solid curve) expression values for single held-out genes in the dataset. These genes are RPL7A, NTH1 and GAD1 respectively.

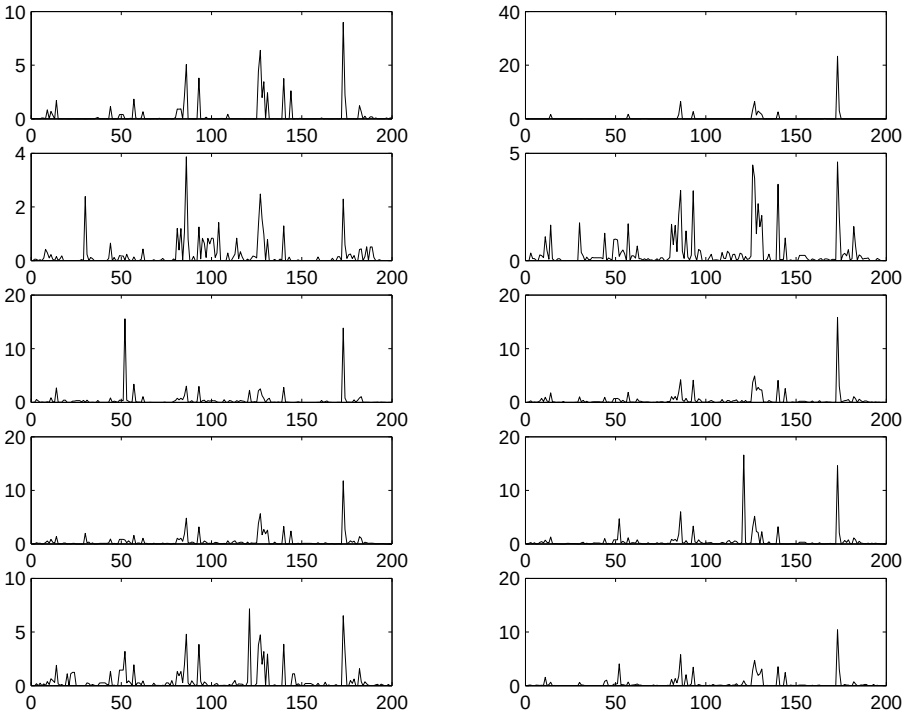


Fig. 5.23: The Poisson parameter β for the Motif to Expression correspondence model **CorrM2E** (note that the scales differ in subplots). One motif (peak) in particular appears significant in all processes.

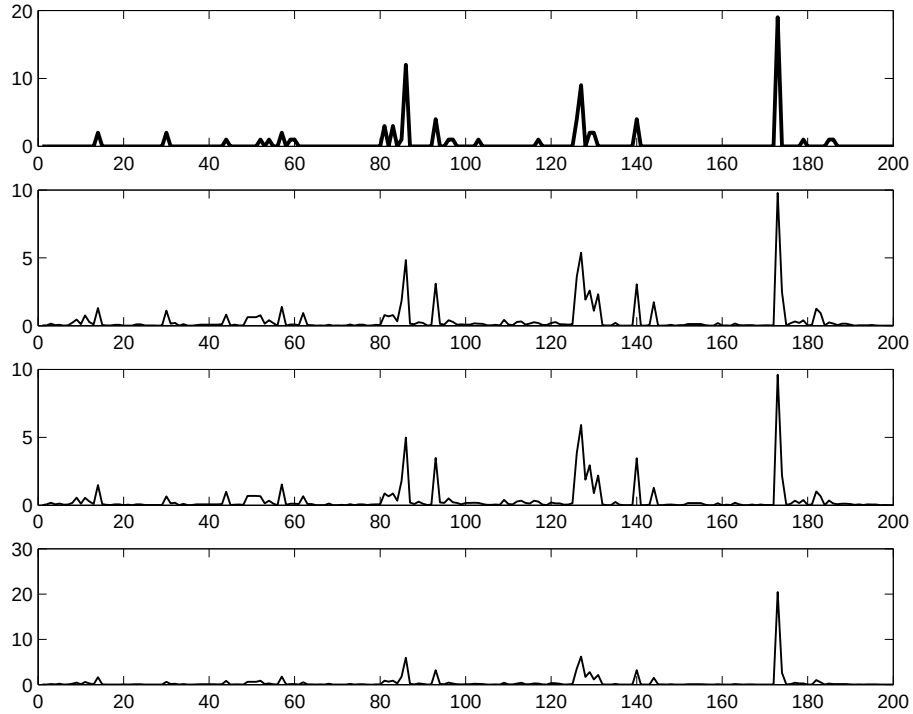


Fig. 5.24: For **CorrE2M** the top subplot shows the actual motif count and the lower three subplots give the principal three processes (probabilities greater than 0.1) from which the algorithm samples in order to predict the motif structure. The probabilities of sampling from these three processes are 0.19,0.14 and 0.31 in descending order.

regulate a particular gene g. In equation (5.7) we would use $\mathcal{G} = (E_g, P_g)$ with P_g the parents. This leads to a corresponding model for integrating knowledge about regulons into the model.

We regard this study as a preliminary step toward construction of a more complex generative probabilistic model for incorporating gene expression, motif and other data types. However, some conclusions already emerge. For example, the likelihood curves in Figures 5.3 and Figures 5.4 have a higher peak for **CorrE2M** than **CorrM2E**. This reflects the fact that there is higher information content in expression data from microarray experiments than there is in the motif data. For the **CorrM2E** model, expression prediction is limited by shortcomings in the motif detection algorithm, because the same transcription factor can simultaneously inhibit expression with one gene while enhancing expression of another and because expression is dependent on subtle differences in the motif structure (cf. Figure 5.14) and location. Of course, probabilistic models are not a unique approach to this problem and one could use established regression methods for prediction. Nevertheless, the probabilistic approach outlined is very flexible since we can investigate prediction of expression from motif data, or indicate the relevance of various motifs given expression data or incorporate other types of data.

Chapter 6

Identification of Prognostic Signatures in Breast Cancer Microarray Data Using Probabilistic Techniques

6.1 Abstract

In this chapter we will apply a new probabilistic data analysis technique (Latent Process Decomposition) to four recent microarray datasets for breast cancer. As this is a probabilistic technique it has a number of advantages over standard hierarchical cluster analysis. These include: An objective assessment of the optimal number of sample or gene clusters in the data via likelihood comparison, an natural inbuilt penalisation of over complex models and a common latent space of explanatory variables for samples and genes. This analysis provides a clearer insight into these datasets, enabling assignment of patients to one of four principal processes. Each of these has a distinct clinical outcome. One process is indolent and associated with under-expression across a number of genes associated with tumour growth. One process is associated with over-expression of *GRB7* and *ERBB2*. The most aggressive process is associated with abnormal expression of transcription factor genes, including members of

the *FOX* family of transcription factor genes, for example. This work was published in - **Journal of the Royal Society** *Interface*, 2005, [20].

6.2 Introduction

Evidence from epidemiological studies, analysis of tumour progression and variability in response to treatment all indicate considerable diversity among human breast cancers. This view is supported by various independent microarray studies [25, 40, 41, 68, 77, 83, 87]. For example, with one recent study [78], hierarchical cluster analysis suggested the existence of five major categories of breast cancer. Two groups of predominantly estrogen receptor positive (ER+) cancers had expression patterns similar to breast luminal cells (called luminal A and B). For the ER− cancers, three additional categories were identified that over expressed genes associated with the *ERBB2* amplicon at 17q22, had a basal cell expression pattern or resembled normal breast tissue. The significantly different clinical outcomes of 4 of these groups (luminal A, luminal B, basal and *ERBB2*) highlighted the potential biological importance of this classification. Although these groups could be broadly defined, the fine structure of dendrograms varied between individual cluster analysis methods and the authors concluded that the observed high level branching was not always a reflection of biologically meaningful relationships.

In this chapter we will apply a new probabilistic approach for finding informative structure in such datasets. This approach is called Latent Process Decomposition (LPD) [70]. It is essentially equivalent to the Latent Dirichlet Allocation approach of Blei *et al* [15] but with Multinomial distributions begin replaced by Gaussian distribution. In the model each sample (or gene expression measurement) is represented as a combinatorial mixture over a finite set of latent processes (a *process* is an assumed functionally related set of samples or genes). Observations are not necessarily assigned to a single cluster. This reflects a prior belief that a number of processes could contribute to a given gene expression level or that a tumour could have a heterogeneous structure because it overlaps several defined states. Most cluster analysis methods use such an implicit mutual exclusion of classes assumption and several algorithms which avoid this, potentially unwarranted, assumption have been proposed recently [34, 63, 18]. The proposed approach has other advantages. For example,

an estimate of the optimal number of sample or gene clusters can be objectively assessed by cross validated likelihood comparison. Also samples and gene expression levels are modelled using a common space of explanatory variables. This is in contrast to the use of dendrograms where samples and gene expression values are typically clustered separately, amounting to two distinct reduced space representations which are not easily related. As a consequence of its probabilistic approach LPD can also readily handle missing values. It has been shown that LPD also compares favourably to various other cluster analysis methods [70].

To illustrate its potential we apply this approach to breast cancer datasets from Sorlie *et al* [78], West *et al* [87], van 't Veer *et al* [83] and de Vijver *et al* [25]. The method appears to give clearer insights into these datasets suggesting at least 4 principal processes, each associated with a different clinical outcome. The method is outlined in Appendix 1.

6.3 Latent Process Decomposition

Here we shall give a brief overview of Latent Process Decomposition (LPD). For a more detailed description of the method the reader is referred to Rogers *et al.* [70]. One of the assumptions of LPD is that each sample can be represented as a combinatorial mixture over multiple processes. This is in contrast to the implicit mutual exclusion of classes assumption of other cluster analysis methods. We have used the term *process* rather than *cluster* to emphasis this difference with standard cluster analysis methods. This also emphasises the the generative process nature of a probabilistic approach. The graphical model for LPD was discussed in the graphical models section 2.2 of chapter 2.

To remind the reader, in the standard way that was given in chapter 2 we will construct a generative probabilistic model, with parameters $\mathbf{\Theta}$ for some data $\mathcal{D}$. We will then maximise the posterior probability of a model parameters given the data, $p(\mathbf{\Theta}|\mathcal{D})$, which from Bayes rule can also be written:

$$p(\mathbf{\Theta}|\mathcal{D}) \propto p(\mathcal{D}|\mathbf{\Theta})p(\mathbf{\Theta}) \tag{6.1}$$

where $p(\mathcal{D}|\mathbf{\Theta})$ is the *likelihood* and $p(\mathbf{\Theta})$ is the *prior* distribution of the model parameters $\mathbf{\Theta}$.

The approach we now outline is described in more detail elsewhere [70]. It is based on the Latent Dirichlet Allocation (LDA) approach to data modelling [15], comparing favourably in likelihood terms with alternatives such as mixture models [58] and other approaches (see [70]). Under this model we assume that the (logged) gene expression ratios from a microarray experiment follow approximate Gaussian distributions. We shall denote the set of gene expressions for an single sample d by $\mathbf{E}_d$, the expression for a single gene as E_{dg} and take $k = 1, \ldots, \mathcal{K}$ as an index for processes. The likelihood of our model is then given by equation 6.2

$$\begin{aligned} \log p(\mathcal{D}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}) &= \sum_d \log \int_\theta p(\mathbf{E}_d|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{\alpha}) d\boldsymbol{\theta} \\ &= \sum_d \log \int_\theta \left[\prod_g \sum_k p(E_{dg}|\mu_k, \sigma_k)\theta_k \right] p(\boldsymbol{\theta}|\boldsymbol{\alpha}) d\boldsymbol{\theta} \end{aligned} \tag{6.2}$$

Exact inference for 6.2 is intractable. We can however lower bound this expression using Jensen's inequality and perform inference using a variational EM algorithm. Thus our approach parallels the Latent Dirichlet Allocation method of Blei et al [15] which derives a similar lower bound for discrete data.

A lower bound on equation 6.2 is constructed through the introduction of two variational distributions $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$ with parameters ϕ_{dgk} and γ_{dk}. $\boldsymbol{\phi}$ is a discrete distribution and $\boldsymbol{\gamma}$ a sample specific Dirichlet distribution. The bound is then maximised for all model parameters and latent variables to give the following update equations:

$$\phi_{dgk} = \frac{\mathcal{N}(E_{dg}|\mu_{gk}, \sigma_{gk}) \exp\left[\psi(\gamma_{dk})\right]}{\sum_{k'=1}^{\mathcal{K}} \mathcal{N}(E_{dg}|\mu_{dk'}, \sigma_{gk'}) \exp\left[\psi(\gamma_{dk'})\right]} \tag{6.3}$$

$$\gamma_{dk} = \alpha_k + \sum_{g=1}^{\mathcal{G}} \phi_{dgk} \tag{6.4}$$

where $\mathcal{N}(\ldots)$ is a normal distribution and $\psi(z)$ is the digamma function. For gene g and process k, μ_{gk} and σ_{gk} are the means and standard deviations (for example, in Figure 6.6 these give the means and spreads for the 4 processes illustrated). γ_{gk}, is the parameter of a variational Dirichlet distribution. From equation ... we see that normalising γ_{gk} or k will give the expected number of times process k was selected in the generation of sample g. The model parameter updates are:

$$\mu_{gk} = \frac{\sum_{d=1}^{\mathcal{D}} \phi_{dgk} E_{dg}}{\sum_{d=1}^{\mathcal{D}} \phi_{dgk}} \tag{6.5}$$

$$\sigma_{gk}^2 = \frac{\sum_{d=1}^{\mathcal{D}} \phi_{dgk} (E_{dg} - \mu_{gk})^2}{\sum_{d'=1}^{\mathcal{D}} \phi_{dgk'}} \tag{6.6}$$

As there is no closed form update for the Dirichlet model parameter α_k and second order gradient descent technique is used (see [15] Appendix A.4.2 and [70]).

The maximum likelihood approach given above can be easily extended to a maximum posterior (MAP) solution. We shall endow the model parameters with suitable prior distributions. Thus, a suitable prior on the means μ would be a Gaussian distribution with zero mean (see section 2.1.3. This would reflect a prior belief that most genes will be uninformative and will have logged expression ratios around zero (i.e. they are unchanged compared to a reference sample). For the variance, we may wish to define a prior that penalises over-complex models and avoids over-fitting. Microarray data is inherently noisy and so a Gaussian which is collapsing onto a single point is highly unlikely. With a suitable choice for the prior an extension of our model to a full MAP solution is straightforward. Our combined likelihood and prior expression is (assuming a uniform prior on $\boldsymbol{\alpha}$):

$$p(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}|\mathcal{D}) \propto p(\mathcal{D}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}) p(\boldsymbol{\mu}) p(\boldsymbol{\sigma}). \tag{6.7}$$

Taking the logarithm of both sides we see that the maximisation task is given by:

$$\boldsymbol{\alpha}, \boldsymbol{\sigma}, \boldsymbol{\mu} = \arg\max_{\boldsymbol{\alpha},\boldsymbol{\sigma},\boldsymbol{\mu}} \quad \log p(\mathcal{G}|\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\alpha}) + \log p(\boldsymbol{\mu}) + \log p(\boldsymbol{\sigma}). \tag{6.8}$$

Thus we can simply append these terms onto our bound on the log-likelihood. Noting that the prior distributions are functions of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ only (and any associated hyper-parameters), we conclude that these extra terms only affect the update equations for μ_{dk} and σ_{dk}. For algebraic simplicity let us assume the following priors:

$$p(\mu_{gk}) \propto \mathcal{N}(0, \sigma_\mu) \tag{6.9}$$

$$p(\sigma^2_{gk}) \propto \exp\left\{-\frac{s}{\sigma^2_{dk}}\right\} \tag{6.10}$$

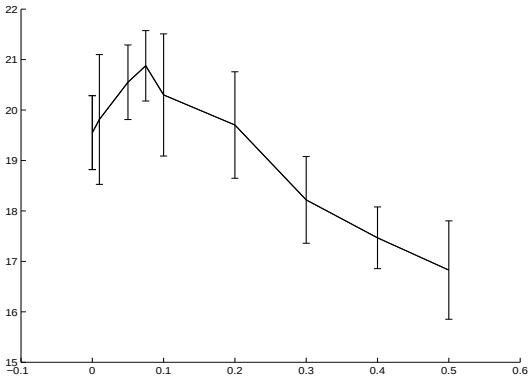
then we obtain the following new update equations:

$$\mu_{gk} = \frac{\sigma^2_\mu \sum_{d=1}^{\mathcal{D}} \phi_{dgk} E_{dg}}{\sigma^2_{gk} + \sigma^2_\mu \sum_{d=1}^{\mathcal{D}} \phi_{dgk}} \tag{6.11}$$

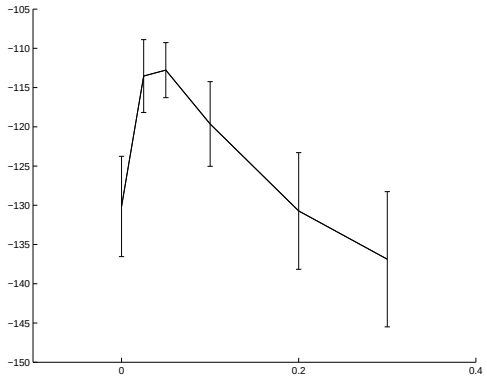
$$\sigma^2_{gk} = \frac{\sum_{d=1}^{\mathcal{D}} \phi_{dgk}(E_{dg} - \mu_{gk})^2 + 2s}{\sum_{d=1}^{\mathcal{D}} \phi_{dgk}} \tag{6.12}$$

The prior for $\boldsymbol{\sigma}$ is improper, in that it does not have a finite integral but as we are only looking for a MAP solution this is acceptable. Once the model parameters have been estimated, we can calculate the likelihood for a collection of $\mathcal{D}'$ samples using:

$$\mathcal{L} = \prod_{d=1}^{\mathcal{D}} \int_\theta \left\{ \prod_{g=1}^{\mathcal{G}} \sum_{k=1}^{\mathcal{K}} \mathcal{N}(E_{dg}|k, \mu_{gk}, \sigma_{gk})\theta_k \right\} p(\boldsymbol{\theta}|\boldsymbol{\alpha}) d\boldsymbol{\theta} \tag{6.13}$$



(a) Sorlie *et al*



(b) van 't Veer *et al*

Fig. 6.1: Hold-out log-likelihood as a function of s for the datasets of Sorlie *et al* (left) and van 't Veer *et al* (right).

where we estimate the expectation over the Dirichlet distribution by averaging over N samples, $\boldsymbol{\theta}_n$ drawn from a Dirichlet with the estimated model parameter $\boldsymbol{\alpha}$, $p(\boldsymbol{\theta}|\boldsymbol{\alpha})$

$$\mathcal{L} \approx \prod_{d=1}^{\mathcal{D}} \frac{1}{N} \sum_{n=1}^{N} \left\{ \prod_{g=1}^{\mathcal{G}} \sum_{k=1}^{\mathcal{K}} \mathcal{N}(E_{dg}|k, \mu_{gk}, \sigma_{gk}) \theta_{kn} \right\} \tag{6.14}$$

We shall use the estimated likelihood from equation 6.14 in a cross validation to both determine the optimal number of processes to use and to determine optimal hyper-parameters used in the prior. Indeed cross validation plots for the hyper-parameter s for two of the data sets we will study are given in figure 6.1. As reported elsewhere [70] the model is little affected by choice of the prior parameter σ_μ in equation (6.9) and we have set this value to 0.1.

6.4 The Application of Latent Process Decomposition to four Microarray Datasets for Breast Cancer

On all the data sets given below we shall perform a 10 fold cross validation of the likelihood. That is we will select 10% and then estimate the model parameters on the remaining 90% of data by iteratating the update equations given in section 6.3. The average likelihood of the left out data will then be estimated using equation 6.14. This is then repeated for each remaining 10% giving us 10 values of the held out likelihood. This data will be used to plot the subsequent likelihood curves.

6.4.1 Data set of Sorlie *et al*

The first dataset which we will use is from the study of Sorlie *et al* [77]. We took data from 115 primary breast carcinoma samples (labelled Norway/Stanford and very predominantly of invasive ductal type) and we used the same set of 552 genes selected in their study. In Figure 6.2 we give both the logged maximum-likelihood curve and logged maximum a posterior curve for a cross validation of LPD [70]. For the maximum likelihood model the log-likelihood has a peak at approximately 4 processes indicating this is a suitable number of processes to use. For the MAP solution (Figure 6.2, upper curve) each model parameter has been given a prior. This cure rises to a plateau after which no further gain is to be made by introducing further processes since the model will not exploit this extra freedom. In contrast, for the maximum likelihood solution, the log-likelihood falls as further processes are introduced since the algorithm will use these and construct an over-complex model.

Taking the choice of a 4 process model and running the update equations of section 6.3 on the whole dataset we can plot then plot figure 6.5. This is a plot where the peaks indicate the expected proportion of genes in a sample d that have been generated by selecting process k (these peaks are given by normalised γ_{dk} parameters). Unlike most cluster analysis methods, samples can belong to several processes simultaneously.

By cutting the plot 6.2 at 0.5 we assign each sample d to at most one process k (indeed some patients may not be assigned to any process) we can determine the corresponding Kaplan-

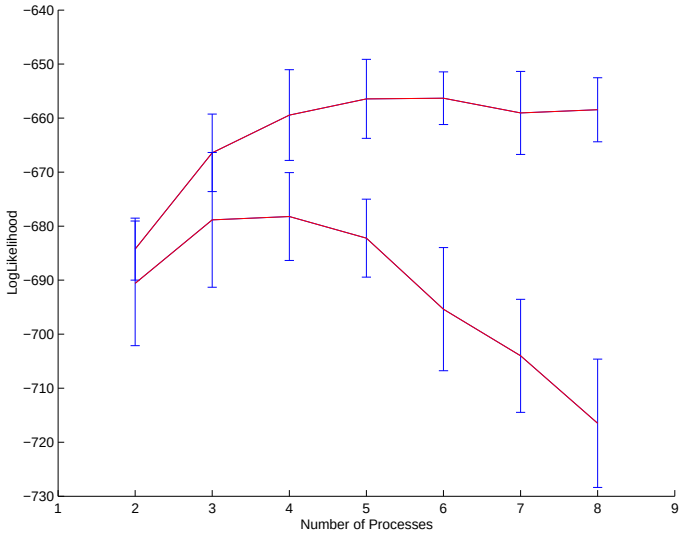


Fig. 6.2: The log-likelihood (y-axis) versus number of processes (x-axis) using the MAP solution (upper curve) and maximum likelihood solution (lower curve) for the Sorlie *et al* dataset Stanford/Norway dataset [77].

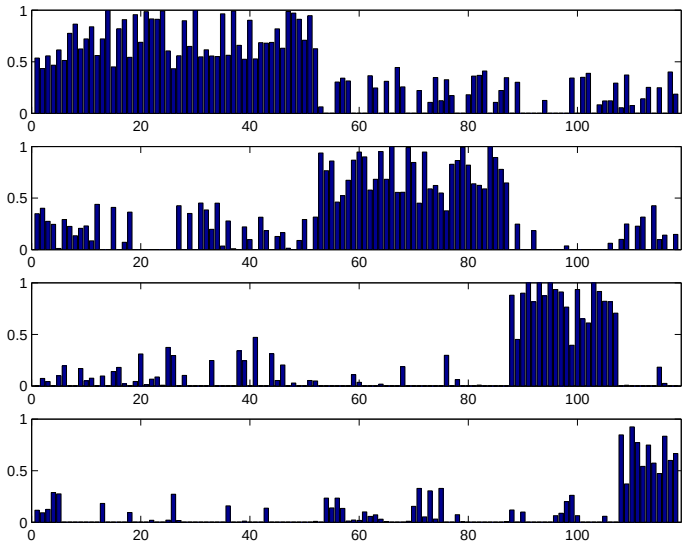
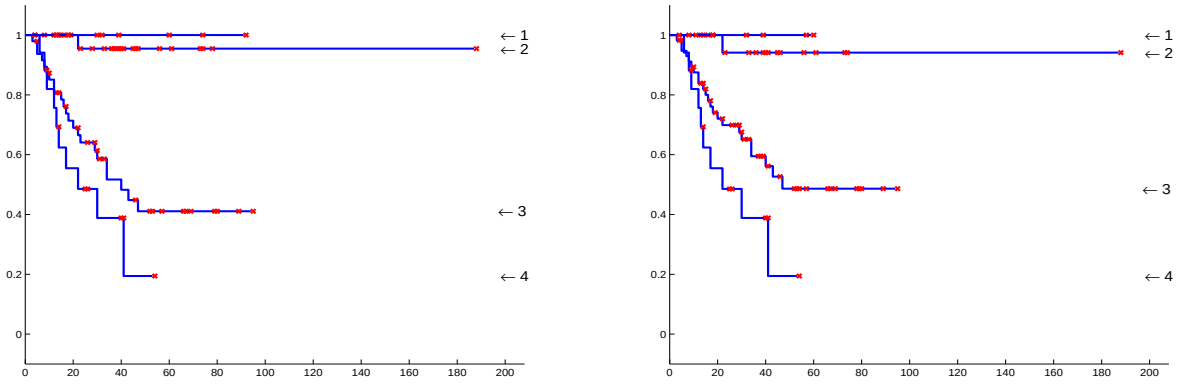


Fig. 6.3: Decomposition diagram derived from LPD for the dataset of Sorlie *et al*. The top process is identified with the trend curve 3 in Figure 6.4(a), the second process is identified with 2, the third with 4 and the lowest is identified with the indolent process 1 in Figure 6.4(a).



(a) KM1

(b) KM2

Fig. 6.4: Kaplan-Meier plots for the Sorlie *et al* dataset. The graphs show fraction not expired from the disease (y-axis) versus number of months (x-axis). For KM1 (left) there are 9 patients in process 1, 32 in 2, 48 in 3 and 18 in 4 (the remaining 8 samples are insufficiently identified with a process). A vertical drop indicates expiry from the disease and a star indicates the patient is not recorded as expired from the disease (this includes the point at which some patients exited the survey). KM2 corresponds to a different initialisation of the algorithm.

Meier plot in Figure 6.4(a). The separation of patients into distinct survival groups is more clear cut than that made by the original authors [78] with one indolent subtype and three aggressive subtypes indicated.

As a consequence of the variational EM algorithm used solutions for model parameters correspond to local maxima in the likelihood space. In particular, local maxima correspond to models with good fits to the data with the intervening regions in model space corresponding to poorer fits. Nevertheless, it is likely that models with good fits are often concentrated in model space. However, this does mean different initialisations of the algorithm will give different solutions. In fact, since many peaks in Figure 6.5 are near 0.5, the Kaplan-Meier plot is the most sensitive result dependent on this effect. Figure 6.4(b) is a typical result from a different initialisation in which some patients have moved between the outcome trends. To investigate this issue we restarted the algorithm with 50 randomly constructed initialisations and found that 32 of these gave a Kaplan-Meier plot in which no patient had expired from the disease in process 1. Furthermore, these 32 solutions had a distinctly higher average log-likelihood than those solutions with at least one patient expiring from the disease in process

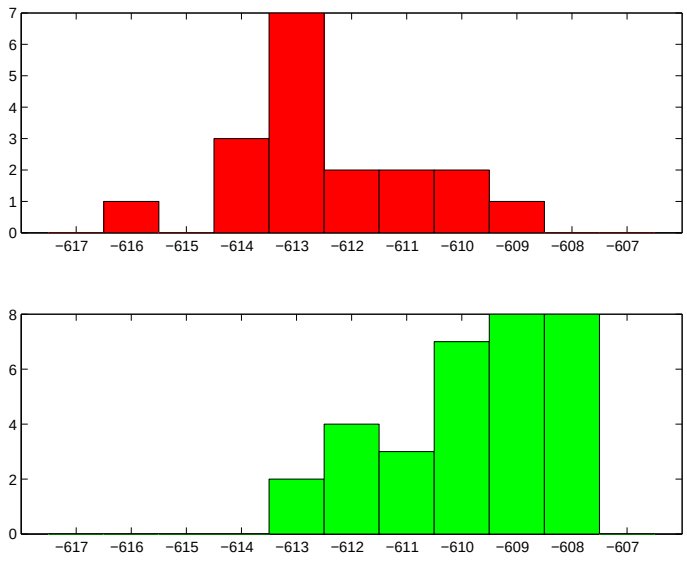
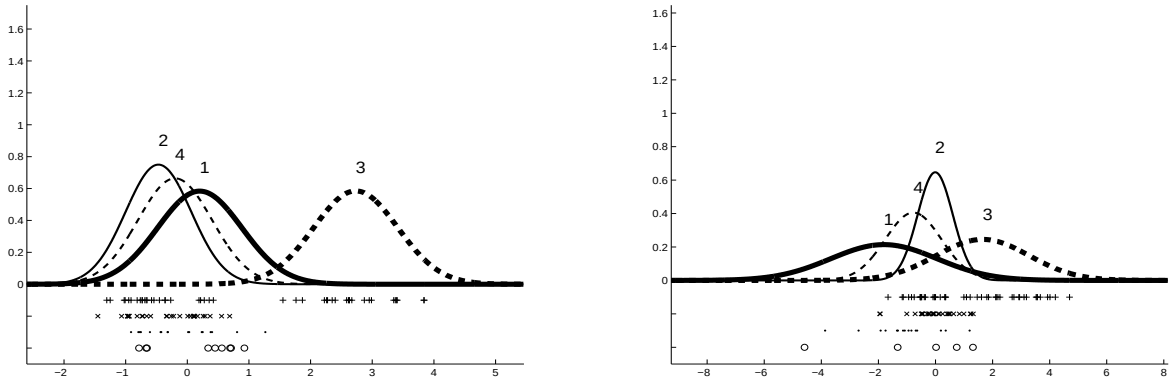


Fig. 6.5: With 50 random initialisations, 32 instances gave Kaplan Meier plots with a purely indolent process 1 (lower histogram) and 18 cases had at least one patient expiring from the disease (upper histogram). The x-axis gives the value of the log-likelihood and the y-axis the frequency of occurrence. Solutions with a purely indolent process 1 gave a higher average log-likelihood indicating they give a better fit to the data.

1, indicating they are more appropriate models (Figure 6.5).

Apart from identifying samples with processes, we can use the model parameters to identify those genes which are most prominent in distinguishing processes. For each gene g in each process k we have determined a mean μ_k and standard deviation σ_k. We can plot density curves for each of these distributions as show how they match the spread of data, $\mathbf{E}_g$ across the whole data set. An example of two density curves is given in Figures 6.6(a) and 6.6(b). These density curves are derived from the dataset taken as a whole and are not one-dimensional fits to the expression values for that gene. To rank genes that distinguish between processes, k_1 and k_2, we can use the score $Z_1 = |\mu_{k_1} - \mu_{k_2}|/\sqrt{\sigma_{k_1}^2 + \sigma_{k_2}^2}$. Apart from comparing two processes we could also compare one process with the rest e.g. by using the lowest pairwise Z_1-score. Unfortunately this score can be adversely influenced by large variances. Thus the gene depicted in Figure 6.8(a) does not score well because it has a large variance in the denominator of Z_1. Consequently we will also use a second, non-parametric rank-based, score (based on the Mann-Whitney test [69]) to highlight such cases. This score will be denoted Z_2 and quantifies the probability of observing a sequence of ranked and labelled data points



(a) GRB7 (b) ERBB2

Fig. 6.6: Inferred densities for *GRB7* and *ERBB2* for the Sorlie *et al* dataset, with + the expression values for samples identified with process 3. Though only over-expressing in process 3 a subset of samples do not over-express *GRB7* suggesting a possible subprocess within this process. In this and subsequent figures individual expression values are marked $\circ$ if the samples are associated with process 1, $\times$ with 2, + with 3 and $\cdot$ if associated with process 4.

(ranked by expression ratio and labelled 1 (process of interest) or 2 (other processes)).

No single gene is a particularly distinct marker for process 1. However, of the top 20 ranked genes distinguishing process 1 from the rest, all but one exhibit relative under-expression in process 1. For the three aggressive processes (2-4), process 4 has the most distinctive genes and process 2 the least distinctive (the highest ranked gene is *LIV-1*). Using the Z_1-score the most distinctive gene in process 3 is *GRB7*, depicted in Figure 6.6(a). It has a score $Z_1 = 3.84$ ($p = 0.00006$) with only $Z_1 = 1.59$ ($p = 0.06$) for the next highest ranked gene (*PAPSS2*). *GRB7* is an adaptor-type signalling protein which is recruited via its SH2 domain to a variety of receptor tyrosine kinases (RTKs), including *ERBB2* and *ERBB3*. It is over expressed in breast, esophageal and gastric cancers, and may contribute to invasiveness potential [67]. It is frequently co-amplified with *ERBB2* (*HER2*) in breast cancer and from Figure 6.6(b) we see that *ERBB2* is, indeed, only over expressed in process 3.

Process 4 has the most distinctive set of genes. In agreement with previous observations [78], this process has basal cell characteristics e.g. cytokeratin 5 appears up-regulated. Using the Z_1 score the top ranked gene distinguishing process 4 is *FLT1* (*VEGFR1*) (Figure 6.7).

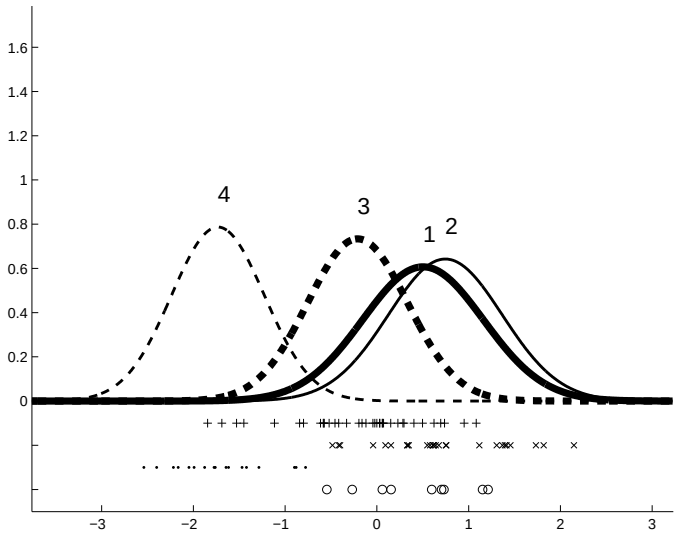


Fig. 6.7: Inferred densities for *FLT1* (*VEGFR1*) in process 4 with · denoting the corresponding expression values.

VEGFR1 (especially its soluble isoform) is a negative regulator of vascular endothelial growth factor availability. Indeed, *VEGFR1* over expression is associated with improved survival in breast cancer [90]. Estrogen mediated decrease in *VEGFR1* expression can cause increased angiogenesis leading to enhanced breast tumour progression [31].

The second ranked gene by Z_1-score is *MAFG* which is associated with up-regulation of protective anti-oxidant enzymes under cellular conditions of oxidative stress [47]. Third ranked is *FOXC1*, a gene which expresses a forkhead transcription factor. The fourth ranked gene is *XBP1* expressing an X box binding protein and the fifth ranked gene expresses *AD021* protein. In the table below we list the top 12 probes ranked by the Z_2 score for process 4.

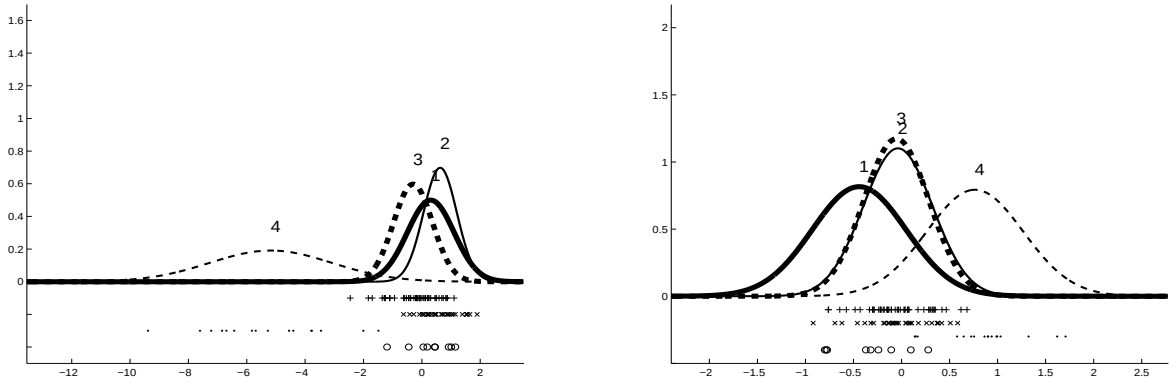
FOXA1 and *FOXC1* are members of the forkhead family of transcription factor genes (Figure 6.8).

FOXA1, *GATA3* and *XBP1* encode transcription factors and their roles and association with the estrogen receptor-α gene (*ESR1*) and trefoil factors (*TFF3* and *TFF1*) are reviewed by Lacroix and Leclerq [50].

In Figure 6.9 we have given the original dendrogram decomposition reported in Sorlie *et al*

Rank	Gene	Z_2-score	Expression
1.	*TFF3*	6.35	Under
2.	*FOXC1*	6.32	Over
3.	*FOXA1*	6.30	Under
4.	*XBP1*	6.25	Under
5.	*GATA3*	6.11	Under
6.	*B3GNT5*	6.08	Over
7.	*FLJ14525*	6.05	Over
8.	*FLT1*	6.04	Under
9.	*GALNT10*	5.95	Under
10.	*FOXC1*	5.88	Over
11.	*FBP1*	5.76	Under
12.	*GATA3*	5.68	Under

Tab. 6.1: The top ranked genes distinguishing process 4 by Z_2-score for the dataset of Sorlie *et al.* Z_2 follows a normal distribution with $\mathcal{N}(0,1)$ thus the associated probabilities of occurrence are upper bounded by 10^{-8} reflecting the fact that the ordering of expression values for process 4 against the set of expression values for the other processes is highly improbable according to a null hypothesis. In the original data the *FOXC1* clone is annotated as *FLJ11796* and *FOXA1* as *HNF3A*.



(a) *FOXA1* (b) *FOXC1*

Fig. 6.8: *FOXA1* (*HNF3A*) under expresses while *FOXC1* over expresses in process 4 ($\cdot$ denotes the expression values in process 4).

[78] along with the assignment to processes given in Figure 6.5. Sorlie *et al* [78] labelled a subset of the tumour samples as Luminal A and B, ERBB2+ and Basal. Their 18 Basal tumours match the 18 Process 4 samples. Indeed, we shall later see that this process is very distinctive. Elsewhere LPD labels a wider range of samples than labelled by Sorlie *et al* (though this would depend on the threshold chosen for the significance of the peaks in Figure 6.5). Their 11 Luminal B and 11 ERBB2+ are exclusively subsets of process 3, while their 28 Luminal A are exclusively associated with processes 1 and 2. Indolent process 1 is exclusively sampled from some Luminal A samples and other samples which were left unlabelled in their study.

6.4.2 Dataset of West *et al*

For the Affymetrix breast cancer dataset of West *et al* [87] we used data from 49 samples (exclusively derived from tumours of invasive ductal type) with 500 genes ordered by their variance across the whole data set. As well as having a high ranking variance genes were only selected if at least 30% of them (15 samples) were clean experimental measurements. This was taken as having a p-value no greater than 0.001. LPD could computationally handle the full dataset but some feature selection is advisable since redundant information clouds analysis with noise. No survival data was available for this dataset, though time-to-metastasis was available. Nevertheless we can derive the corresponding MAP solution which also plateaus after 4 processes (Figure 6.10).

Using 4 processes, we then get the following decomposition diagram given in figure 6.11.

As observed previously, process 4 has the most distinctive genetic signature which, from time-to-metastasis data, appears identified with the second row in Figure 6.11. The top-ranked genes distinguishing this process are given in the Table 6.2.

Interestingly, *GATA3*, *FOXA1*, *XPB1*, *TFF3* and *FPB1* are in common between this Table and Table 6.1. Though *GRB7* and *ERBB2* were highlighted previously [78] they were not selected in the 500 genes of choice as they did not pass the feature selection. Though this fact most likely stems from the smaller dataset size.

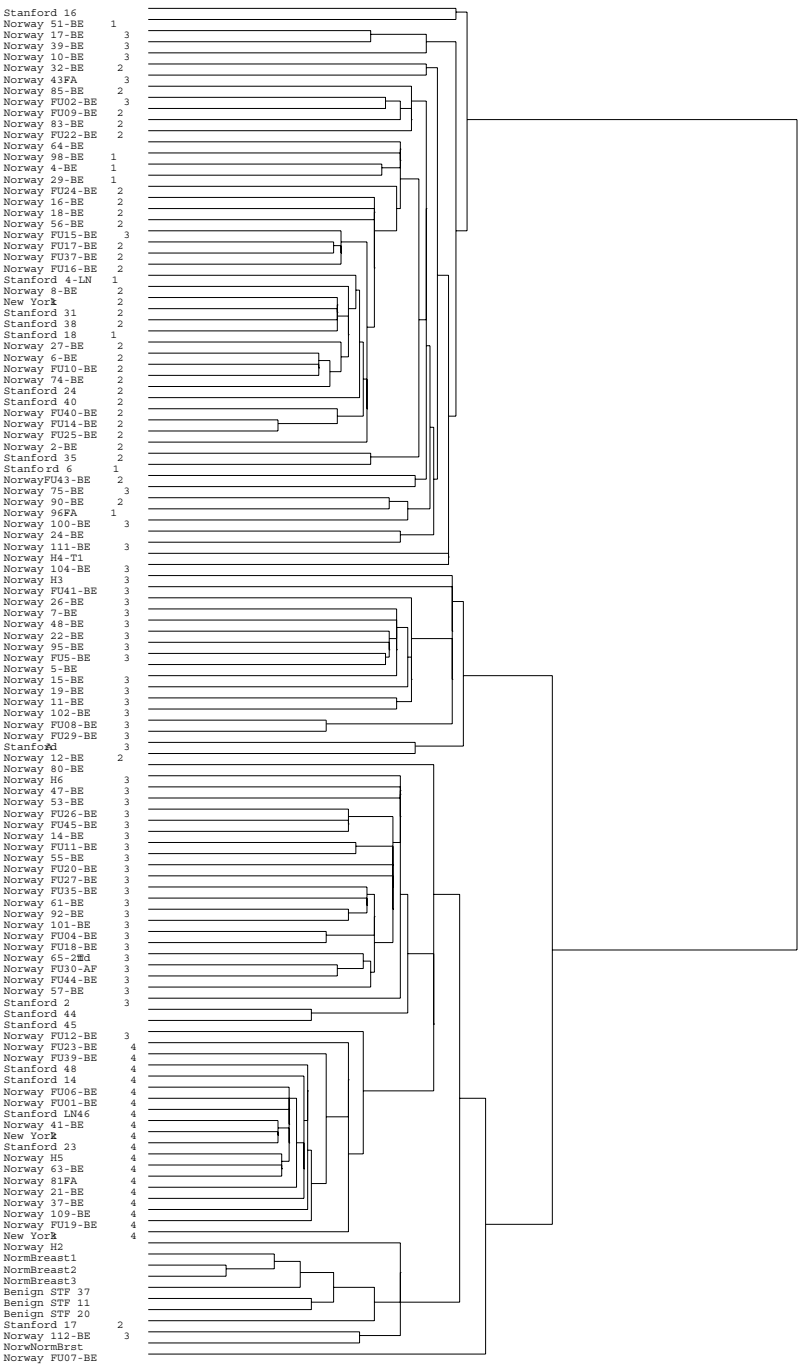


Fig. 6.9: A comparison between the dendrogram reported in Sorlie *et al* [78], Figure 1B, and the decomposition by LPD given here in Figure 6.5. Underneath the tree the LPD assignment to process is designated by the numbers 4 to 1. Below these numbers are sample titles for identification with Sorlie *et al* [78], Figure 1B. Process assignment numbers are missing in a few cases because the peak in Figure 6.5 (normalised γ_{dk}, see equation 6.4, Appendix 1) was ambiguous in its assignment of sample to process)

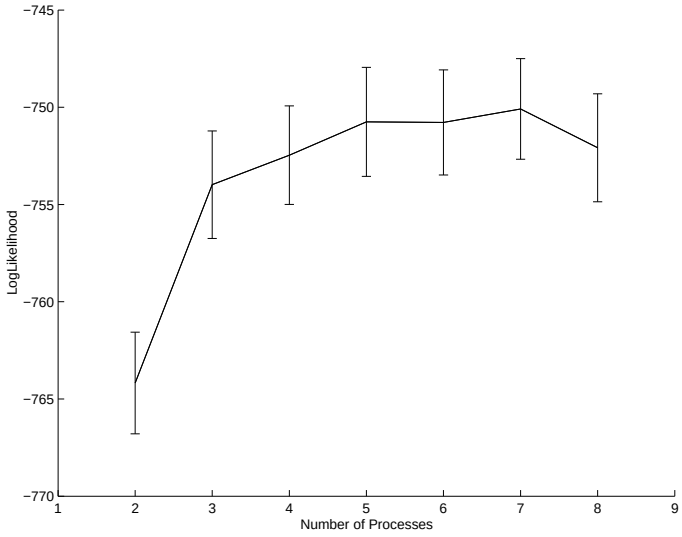


Fig. 6.10: The log-likelihood (y-axis) versus number of processes (x-axis) using a MAP approach (right) for the West *et al* dataset.

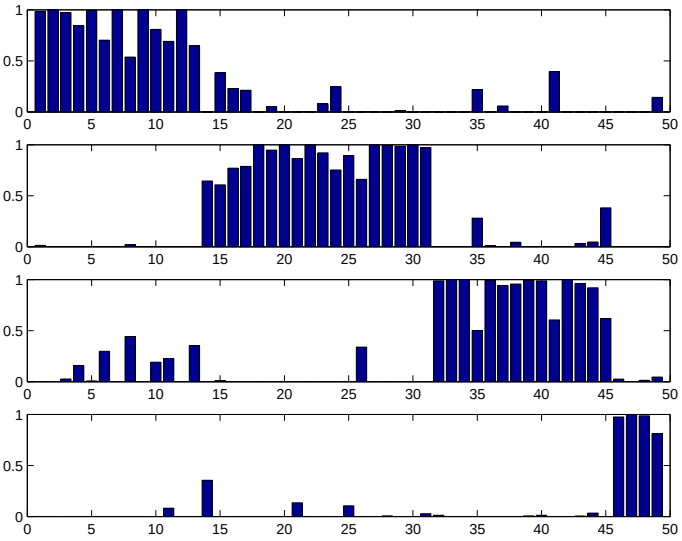


Fig. 6.11: Decomposition diagram derived from LPD for the dataset of West *et al*.

Rank	Gene	Z_2-score	Expression
1.	*hCRHP*	5.51	Under
2.	*XBP1*	5.50	Under
3.	*FOXA1*	5.26	Under
4.	*FPB1*	4.98	Under
5.	*FLJ13710*	4.94	Under
6.	*GATA3*	4.94	Under
7.	*GATA3*	4.92	Under
8.	*CNAP1*	4.90	Over
9.	*NFIB2*	4.83	Over
10.	*Human complement factor B*	4.83	Under
11.	*TFF3*	4.79	Under
12.	*FLJ13710*	4.78	Under

Tab. 6.2: Top ranked genes using the Z_2-score distinguishing a tentative process 4. Using the Z_1 score *GATA3* is ranked 2nd, *FOXA1* is 3rd, *XPB1* is 4th and *TFF3* is 6th. The probabilities of occurrence are upper bounded by 2×10^{-6} (for $Z_2 = 4.78$).

6.4.3 Dataset of of van 't Veer *et al*

For the dataset of van 't Veer *et al* [83] we used samples from 78 patients with primary breast carcinomas, a further 18 samples from patients with *BRCA1* germline mutations and 2 samples with *BRCA2* mutations. We used 500 genes selected in the same way to as in the West data in section 6.4.2, using those genes with a p-value of less than 0.001 in more than 30 tumours. Survival data is not available though we can still compute the log-likelihood curves (Figure 6.12) and this suggests a peak at 4 processes.

The spectrum of peaks corresponding to Figure 6.5 indicated that 16 of the 18 *BRCA1* mutation carriers belonged in one process (which, from the time to metastasis data, appeared to be process 4 in Figure 6.4(b)). The other 2 *BRCA1* samples were spread between processes and, interestingly, were the only 2 patients not to proceed to metastasis. The two *BRCA2* samples belonged together in the same process, distinct from the process associated with the *BRCA1* samples. This picture agreed with the interpretation by dendrogram of Sorlie *et al* [78].

Using the Z_1-score, one process has *ERRB2* (Figure 6.13(a)) and *GRB7* (Figure 6.13(b)) in

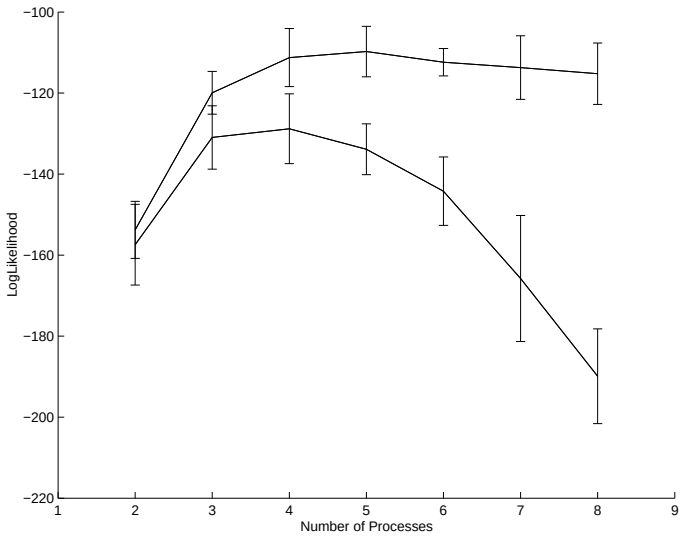


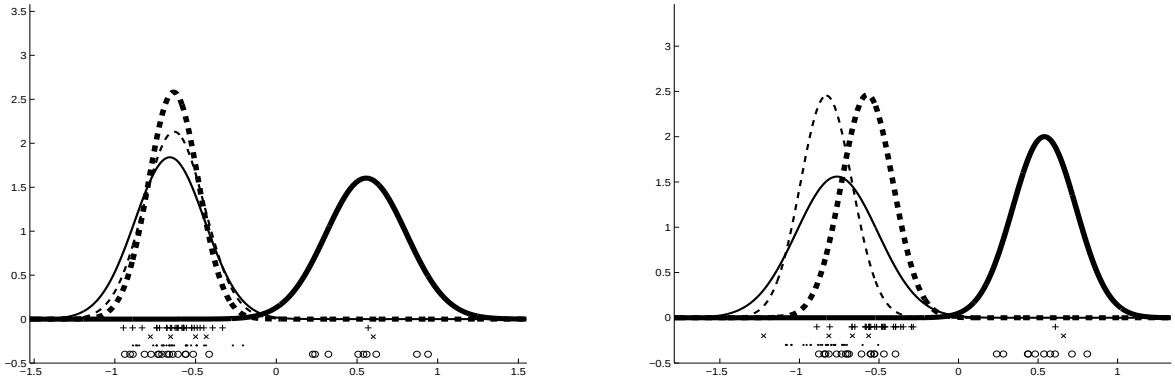
Fig. 6.12: The log-likelihood (y-axis) versus number of processes (x-axis) using the MAP solution (upper, plateauing curve) and maximum likelihood (lower curve) solution for the Van 't Veer *et al* dataset [83].

second and third ranked position with the distribution of expression values having a similar bimodal distribution to that in Figures 6.6(a) and 6.6(b).

The highest ranked Z_2-scores for genes in the four processes are 7.02, 5.85, 5.61 and 2.87. Interestingly, the most distinctive process (with $Z_2 = 7.02$) is associated with genes described previously for process 4, such as *TFF3* and *FOXC1* (Table 6.3). *TFF3*, and the *GATA3*, *FOXA1* and *XPB1* genes mentioned previously, all feature in a small gene expression graph derived from a sparse graphical model [27, 28] indicating genes closely linked with the estrogen receptor gene.

6.4.4 Dataset of de Vijver *et al*

The study of van 't veer *et al* preceded a larger study by de Vijver *et al* [25] which used 295 samples from patients with primary breast carcinomas. The authors of this study discovered tentative signatures for poor and good prognosis using a reduced 70 gene set selected from 24,479. In Figure 6.16 we present a Kaplan-Meier plot with the lower dashed curve corresponding to patients in the poor signature cohort and the upper dashed curve corresponding



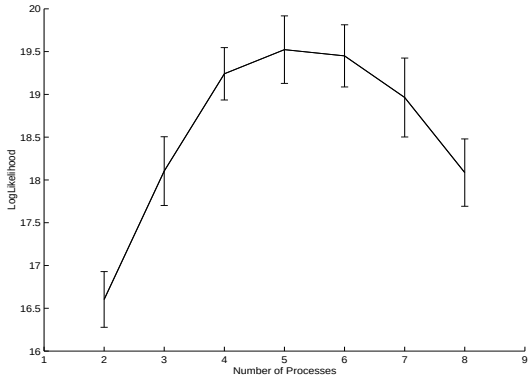
(a) GRB7

(b) ERBB2

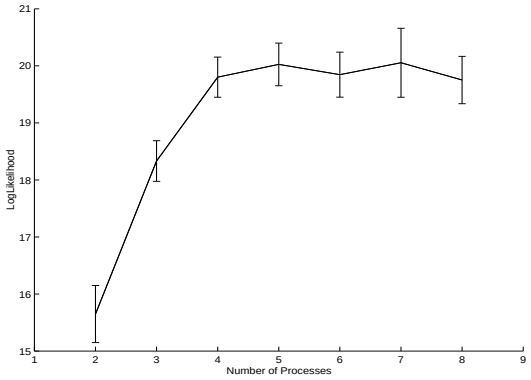
Fig. 6.13: Inferred densities for *GRB7* and *ERBB2* for the dataset of van 't Veer *et al.*

Rank	Gene	Z_2-score	Expression
1.	*TFF3*	7.02	Under
2.	*AGR2*	6.89	Under
3.	*FOXC1*	6.79	Over
4.	*GABA*	6.75	Over
5.	*VGLL1*	6.68	Over

Tab. 6.3: *TFF3* and *FOXC1* are first and third ranked for the most distinctive process in the dataset of van 't veer *et al.* Similarly they are first and second ranked for the most distinctive and aggressive process (4) in the data of Sorlie *et al* (Table 6.1).



(a) Maximum likelihood solution



(b) MAP solution

Fig. 6.14: The log-likelihood (y-axis) versus number of processes (x-axis) using a maximum likelihood and MAP approach for the de Vijver *et al* dataset.

to the good signature cohort. In Figure 6.14(a) we have re-analysed the same dataset (295 samples, 70 features) using LPD and a maximum likelihood approach. The curve shows a peak in the range 4 to 6 processes, implying that the 2-process model proposed by the original authors [25] is a sub-optimal interpretation of the data. In Figure 6.14(b) we see that the likelihood curve for the MAP solution plateaus after using 4 processes.

If we plot the corresponding Kaplan-Meier curves for Figure 6.5 we get the curves in Figure 6.16 in which the top process in Figure 6.5 is identified with curve 3 in Figure 6.16, the second process is identified with curve 4, the third process with 2 and the fourth (lowest) with 1. Compared to the original analysis of de Vijver *et al* (dashed curves in Figure 6.16), all patients in processes 3 and 4 derive from their lower (poor prognosis) group while 10 patients in process 1 are derived from their upper (good prognosis) group and 2 are derived from their poor prognosis group. All patients in process 2 derive from their good prognosis group. Thus our analysis is compatible with their description while enhancing the distinction between clinical outcomes (the solution presented here corresponds to the highest likelihood solution found in numerical experiments).

The inferred densities for two top-ranked genes separating processes 1 and 4 are given in Figures 6.5 and 6.17(b). In fact, of the 26 top-ranked genes separating processes 1 and 4, 21 genes move from under-expression to over-expression as we progress from indolent to the

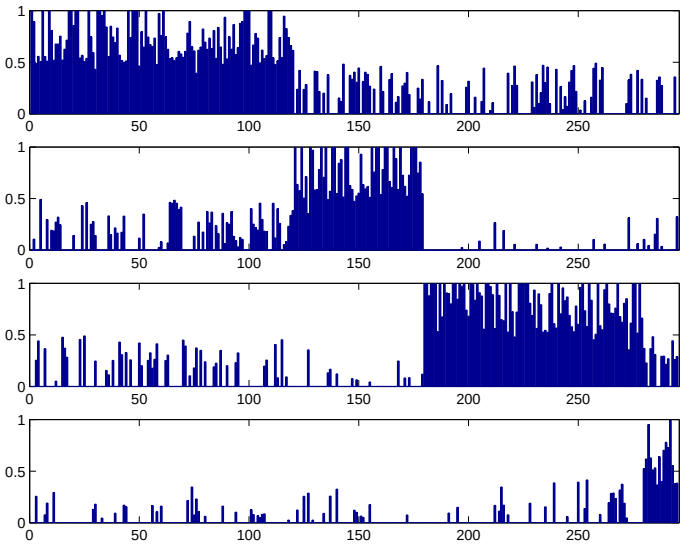


Fig. 6.15: A 4 process decomposition of the data by LPD. The data is not in the same order as the dendrogram.

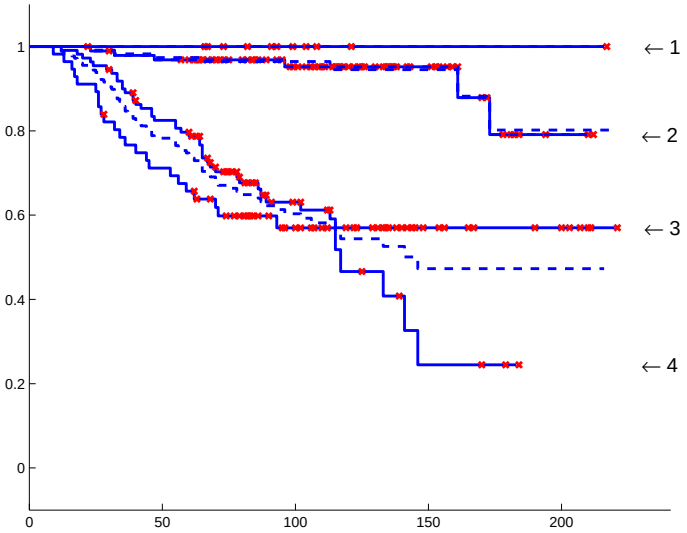
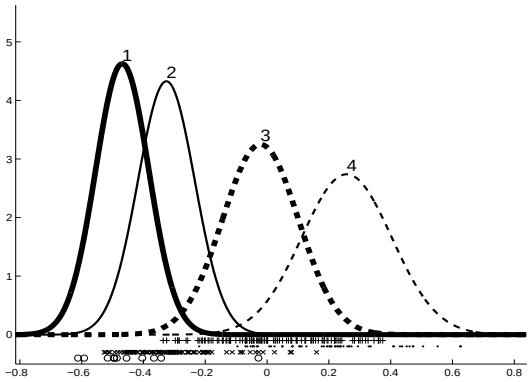
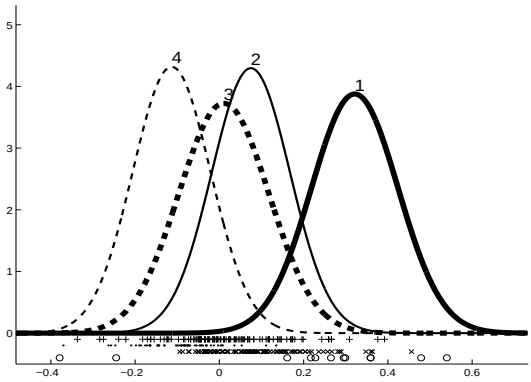


Fig. 6.16: Kaplan-Meier plot for the processes identified in Figure 6.5: fraction not expired from the disease (y-axis), versus number of months (x-axis). The curves labelled 3 and 4 meet at the midpoint *but do not cross over*. The number of patients identified with each curve is 12 (process 1), 97 (2), 110 (3) and 56 (4) (these numbers do not sum to 295 because some samples are ambiguously identified). The original split of de Vijver *et al* [25] are given as dashed curves for comparison.



(a) *ORC6L*



(b) *STK32B*

Fig. 6.17: Inferred densities for *ORC6L* and *STK32B*. The individual expression values are given below the inferred density curves, with ∘ associated with process 1, × with 2, + with 3 and · with process 4.

most aggressive subtype, following the trend in Figure 6.5, while 4 genes follow the reverse trend illustrated in Figure 6.17(b).

The observation that most of the listed genes under-express in process 1 agrees with an observation for the dataset of Sorlie *et al* in which we found that 19 from the top ranked 20 genes distinguishing process 1 from the others under-expressed on the average in process 1. The gene names, their mean expression values per process and this trend are discussed in further detail in Appendix 2 to this paper.

6.5 Monte Carlo Analysis

In this section we apply the Gibbs sampler derived in section 2.4.3 to the data set of Sorlie *et al* and De Vijver *et al*. This will provide a full posterior distribution for the model parameters. We can use this posterior distribution to investigate how accurate the point estimate approximations from the variational EM algorithm are. To implement a Gibbs sampler we shall use the full conditional distributions from equations 2.97 ... 2.101. We have to select hyper parameters for the prior distributions of α, μ and σ^2. The choices here are by no means

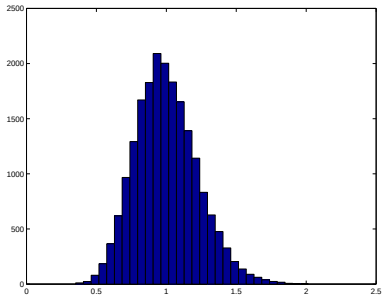
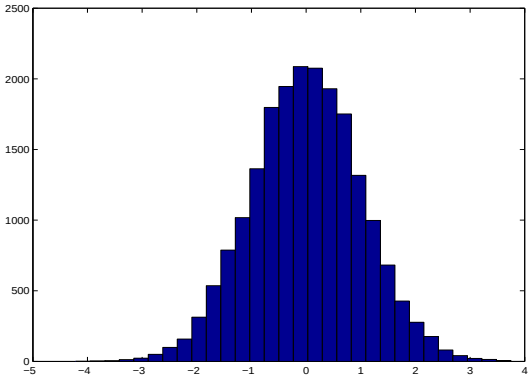
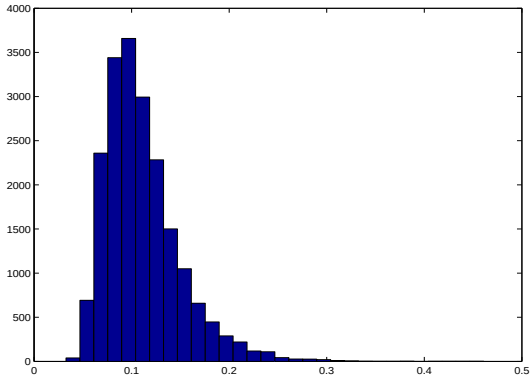


Fig. 6.18: The prior distribution of α is a gamma distribution with parameters $a = 20$ and $b = 0.05$. Note this is unnormalised.



(a) Prior for μ



(b) Prior for σ

Fig. 6.19: Unnormalised prior distributions for the Gaussian parameters.

optimal but express some confidence, based on experience, in the range of values that the parameters are likely to take. The prior on α was chosen to have parameters $a = 20$ and $b = 0.05$ giving a mean of 1 and variance 0.05, a plot of this is given in figure 6.18. The prior on μ is a simple $\mathcal{N}(0, 1)$ and the prior on σ^2 is $InverseGamma$ with parameters 10 and 1 for Sorlie *et al* and 50 and 1 for De Vijver *et al*. The difference in priors on the variance is down to a smaller spread of expression values in the De Vijver *et al* data. Plots of these two distributions for Sorlie *et al* are given in figure 6.19.

Each variable in the algorithm was initialised randomly. There was then a *burn in* period of 40000 iterations to allow the Monte Carlo algorithm to stabilise, then the next 5000 samples were taken to form the posterior distributions. To make a comparison with the results of the variational EM algorithm we chose there to be 4 process, this was in keeping with previous

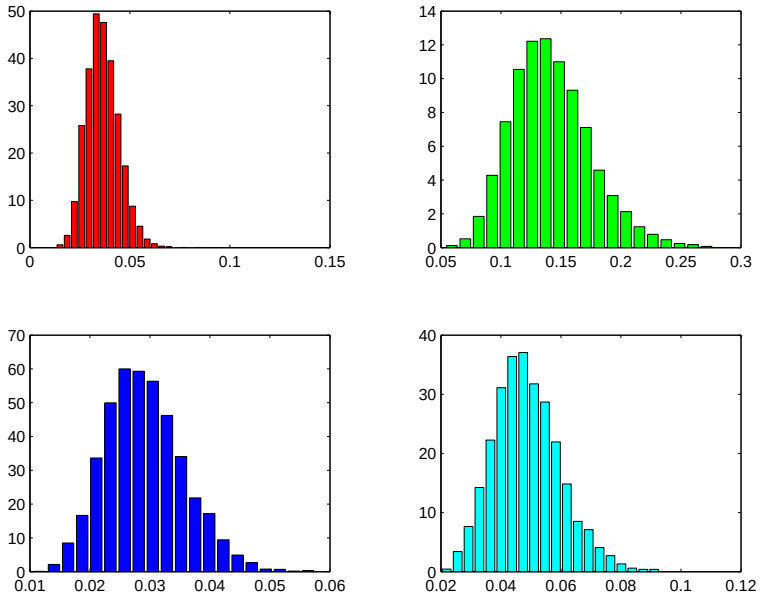


Fig. 6.20: The posterior distribution of the components of α.

analysis. Figure 6.20 shows the posterior distribution for each component of α. The plot indicates sampling from this will generally give quite low values of α, this is equivalent to a peaked simplex for the corresponding Dirichlet distribution.

As a comparison to figure 6.8(a) figures 6.21 and 6.22 show the posterior distributions for the mean and variance of the mixture density for the *FOXA1* gene. It should be stressed that despite obvious similarities figures 6.8(a) and 6.21 are not comparable. Figure 6.21 is plot of the distributions that govern where the peak of each Gaussian in 6.8(a) lies. From this and correspondingly from 6.22 we can see that the point estimate obtained from the variational EM algorithm is a good approximate of the mode of the full posterior.

Similarly, figure 6.23 shows the posterior distribution of the μ for the gene *FLT1*. The distributions of expression for *FLT1* based on the point estimates derived from the variational EM algorithm is given in figure 6.7.

To construct a Kaplan-Meier plot of patient survival we need to assign patients to a single class. Previously this was done by assigning patients based on the γ latent variable associated with their sample. In the Monte Carlo analysis we have a multinomial variable θ which

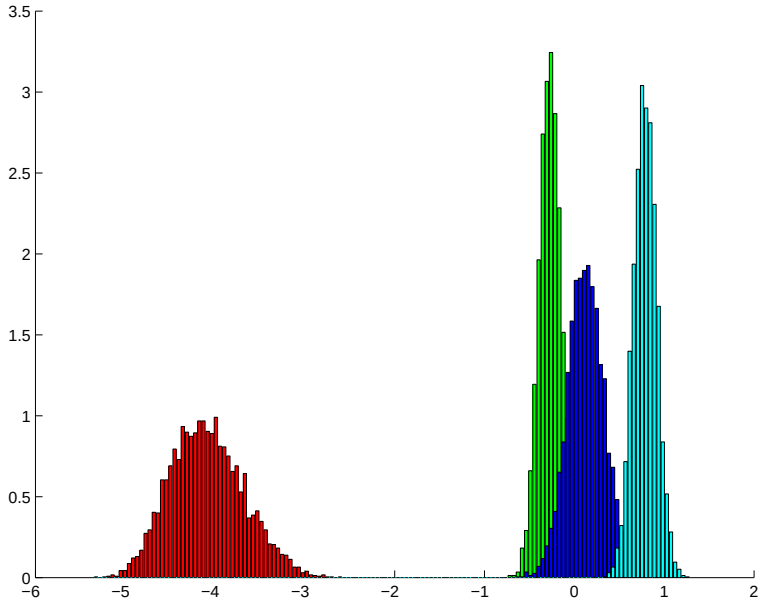


Fig. 6.21: The posterior distribution of μ for *FOXA1* (*HNF3A*).

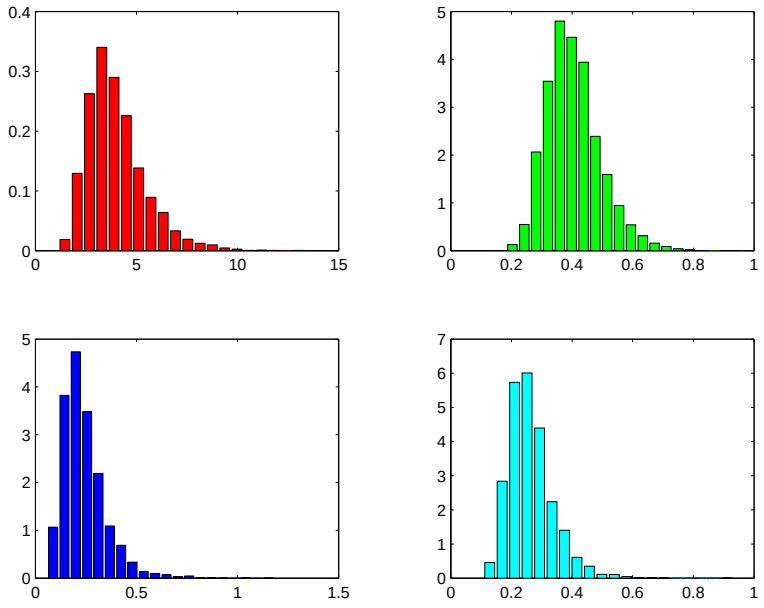


Fig. 6.22: The posterior distribution of σ^2 for HNF3A.

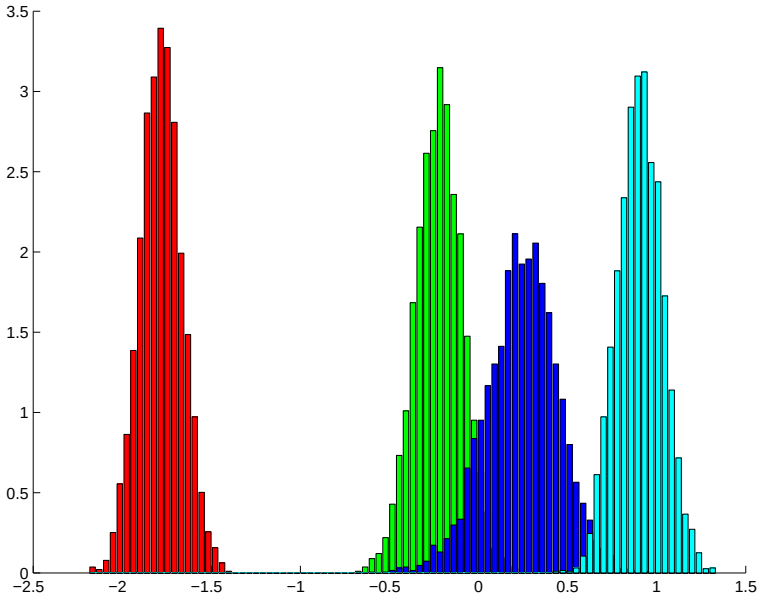


Fig. 6.23: The posterior distribution of μ for *FLT1* (*VEGFR1*).

effectively governs membership to each process. Figure 6.24 shows the distributions of θ across 4 processes for patient 12. By taking a the mode of each of these distributions, normalising these values and then having a threshold of 0.5 we can assign each patient to a process (or indeed no process if there is no clear association).

Using the normalised mode of θ we can derive a survival curve for the patient cohort. The normalised θ plot is give in figure 6.25, as this is for comparison to figure we have ordered the patients (x axis) is the same way. The corresponding survival curve is given in figure 6.26. As the processes are assigned within the algorithm the original ordering of 6.5 has changed in 6.25, with the transitions being $1 \rightarrow 2$, $2 \rightarrow 4$, $3 \rightarrow 1$ and $4 \rightarrow 3$. Apart from the reordering the results are very similar, with the key feature being that in the variational EM plot (figure 6.5) patients are more likely to have a stronger association with one particular process. The apparently more noisy results of the Monte Carlo approach can be put down to the time for which the algorithm was run. One would expect more stable crisper distributions if you allowed a longer *Burn In* period and more samples to be taken.

Again Using the normalised mode of θ we can derive a survival curve for the patient cohort for the dataset of De Vijver *et al.* The normalised θ plot is give in figure 6.27, as this is

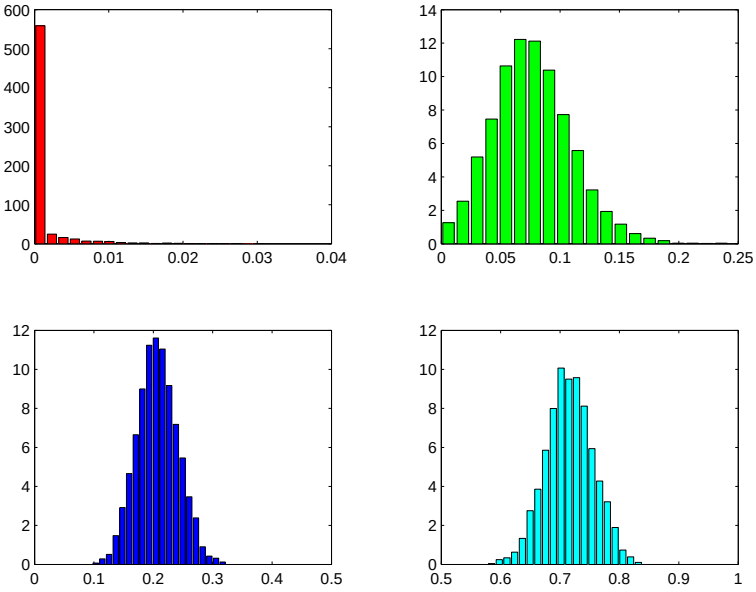


Fig. 6.24: The posterior distribution of θ for Sample 12.

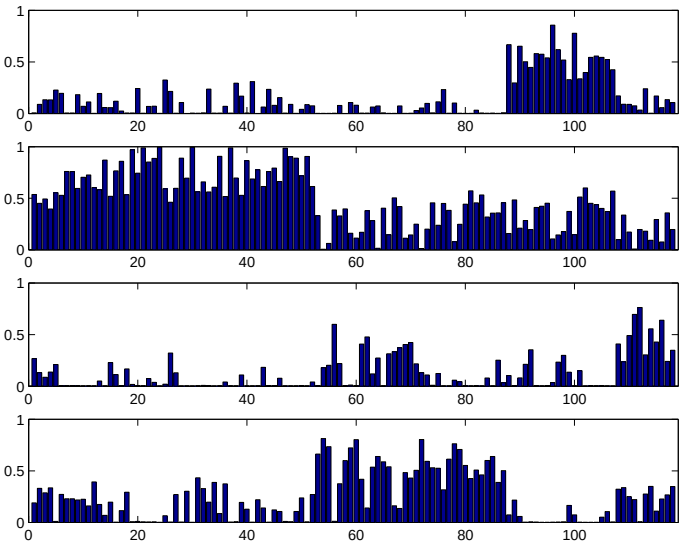


Fig. 6.25: Decomposition diagram derived from LPD for the dataset of Sorlie *et al* using a Monte Carlo approach to inference.

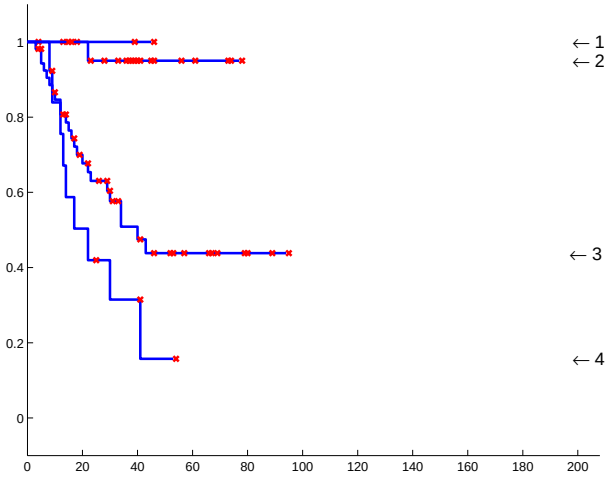


Fig. 6.26: Kaplan-Meier plots for the Sorlie *et al* dataset. The graphs show fraction not expired from the disease (y-axis) versus number of months (x-axis). There are 5 patients in process 1, 23 in 2, 54 in 3 and 14 in 4 (the remaining 19 samples are insufficiently identified with a process). A vertical drop indicates expiry from the disease and a star indicates the patient is not recorded as expired from the disease (this includes the point at which some patients exited the survey).

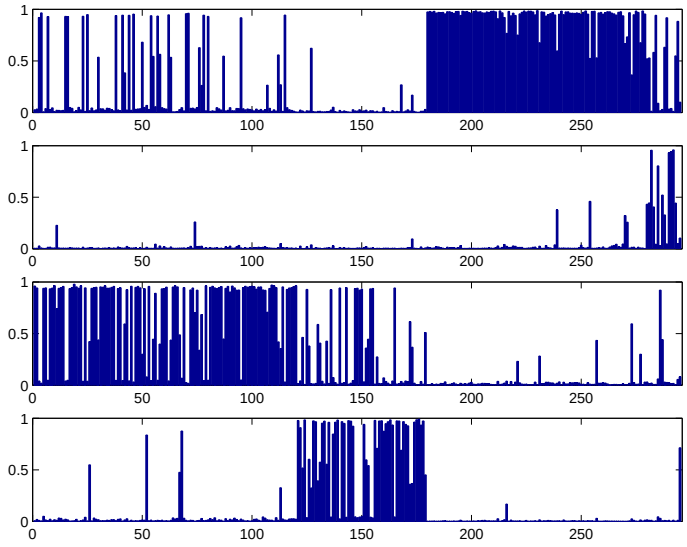


Fig. 6.27: Decomposition diagram derived from LPD for the dataset of De Vijver *et al* using a Monte Carlo approach to inference.

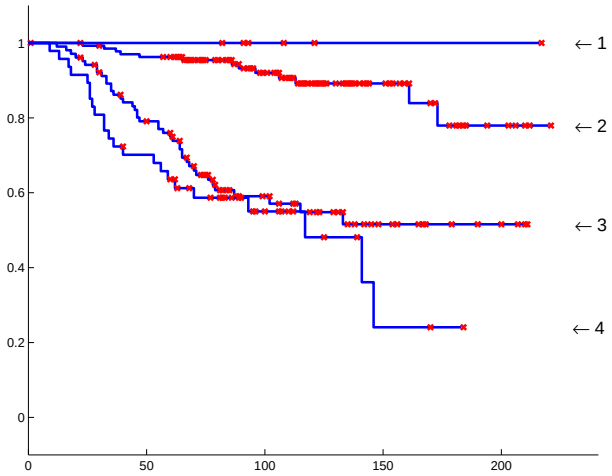


Fig. 6.28: Kaplan-Meier plots for the De Vijver *et al* dataset. The graphs show fraction not expired from the disease (y-axis) versus number of months (x-axis). There are 6 patients in process 1 (2), 136 in 2 (3), 103 in 3 (1) and 47 in 4 (4) (the remaining 3 samples are insufficiently identified with a process and the number in parenthesis is the column in figure 6.27 with (1) top and (4) bottom). A vertical drop indicates expiry from the disease and a star indicates the patient is not recorded as expired from the disease (this includes the point at which some patients exited the survey).

for comparison to figure 6.5 we have ordered the patients (x axis) is the same way. The corresponding survival curve is given in figure 6.28. As the processes are assigned within the algorithm the original ordering of has changed in 6.27, with the transitions being $1 \rightarrow 3$, $2 \rightarrow 4$, $3 \rightarrow 1$ and $4 \rightarrow 2$. Apart from the reordering the results are once again very similar, with the key feature being that in the variational EM plot (figure 6.5) patients are more likely to have a stronger association with one particular process. Interestingly the small indolent subgroup in maintained.

Figure 6.29 shows the posterior distributions for the μ of gene *ORC6L*. A plot of the density of the expression of *ORC6L* using the point estimate derived by the EM algorithm was given in figure . The same progression is seen from under expression for the indolent subgroup through to over expression for the aggressive subgroup.

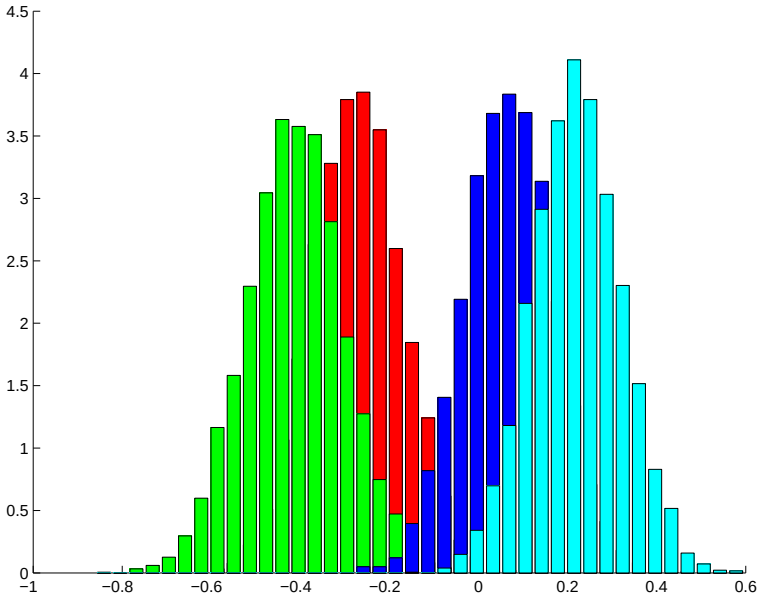


Fig. 6.29: The posterior distribution of μ for *ORC6L*.

6.6 Variational Bayesian Inference

In this section we apply the Variational Bayesian (VB) approach to parameter estimation given in section 6.6. We shall use data sets of Sorlie *et al* and De Vijver *et al* as examples and demonstrate that the results are consistent with those from the EM algorithm. The variational Bayesian approach differs from the EM algorithm as its aim is to estimate the hyper parameters governing the distributions for each model parameter. To proceed with the algorithm we iterate equations 2.55 and 2.56 until convergence. At each iteration we are maximising a lower bound on the evidence of the data (given in equation 2.57). After every iteration we evaluate the bound and continue until there is no significant increase (we took this as a difference of 0.00001, but this is rather arbitrary). As all model variables are *explained away* within VB it is more straightforward to estimate the optimal number of processes present in the data. No cross validation is required and so all the available data can be used. To compare two models, with a different choice of the number of processes K, we can simply compare the final values of bound $F(\boldsymbol{\Theta})$, after convergence. The section is designed to justify the finding of sections where we have used a variational EM algorithm.

The first data set we analysed was that of De Vijver *et al*. Variables were randomly initiated

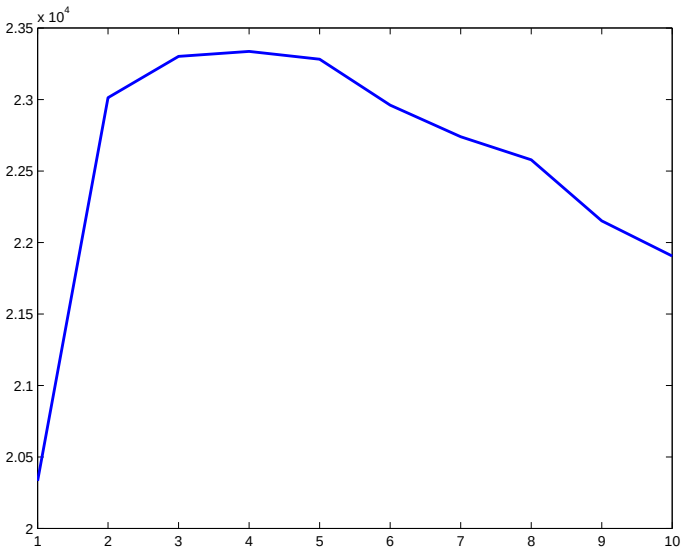


Fig. 6.30: Free energy $F(\boldsymbol{\Theta})$ (a bound lower on the evidence $p(Data|K)$) against K for the data set of De Vijver *et al*

and the prior parameters given in the graphical model defined in figure 2.14 were set to $v_0 = 1$, $m_0 = 0$, $a_0 = t_0 = 20$ and $b_0 = s_0 = 0.05$. A plot of the final bound for different choices of k is given in figure 6.30, this demonstrates that an optimal choice fr k would be between 3 and 5. Referring to the MAP and ML solutions (provided by the EM algorithm) given in figure 6.14 we see the VB solution show a consistent picture for the choice of k. As an analogous exercise to the Kaplan Meier plots given in figure 6.16, we chose 4 processes as optimal and took membership of individual patients to a process as defined by the Dirichlet parameter $\tilde{\boldsymbol{\theta}}$ given in equation 2.55. This is an equivalent variable to γ used in the EM approach. A survival curve based on this criteria is given in equation 6.31. As was seen in figure 6.16, LPD clusters the patients into groupings that have distinct survival properties. This includes an indolent group in which no patients expire and an aggressive group in which the vast majority expire.

Similarly we can perform identical analysis on the data set of Sorlie *et al*. Comparing figure 6.32 to 6.2, again we see an estimate for the number of subtypes as approximately 4. Taking this as the number of processes and generating a KM plot based on the criteria used for the De Vijver *et al*, we generate the figure 6.33. This is consistent with previous plots given in the EM algorithm 6.4(a) and the Monte Carlo based inference, 6.26.

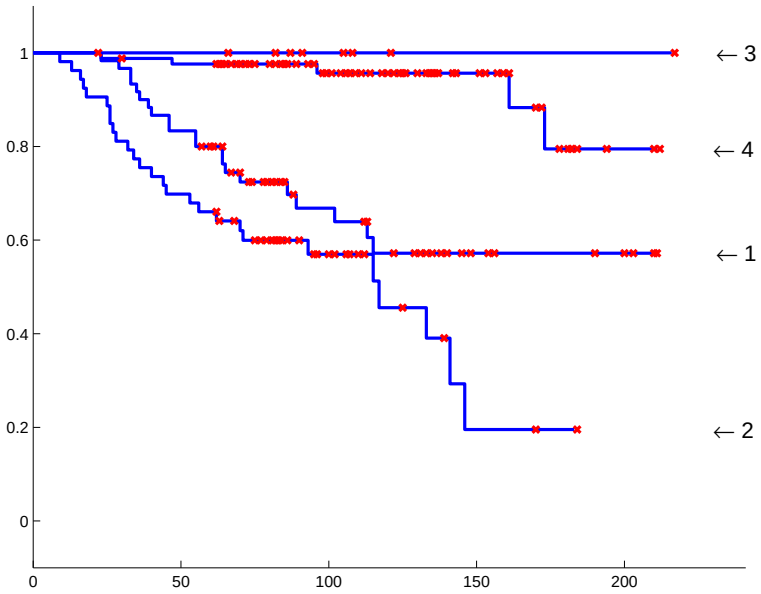


Fig. 6.31: Kaplan-Meier plots for the De Vijver *et al* dataset. The graphs show fraction not expired from the disease (y-axis) versus number of months (x-axis). There are 9 patients in process 1, 85 in 2 , 60 in 3 and 53 in 4 (the remaining samples are insufficiently identified with a process). A vertical drop indicates expiry from the disease and a star indicates the patient is not recorded as expired from the disease (this includes the point at which some patients exited the survey). Note, the survival curves do not cross or touch but merely go too close for the resolution of the image to distinguish them clearly.

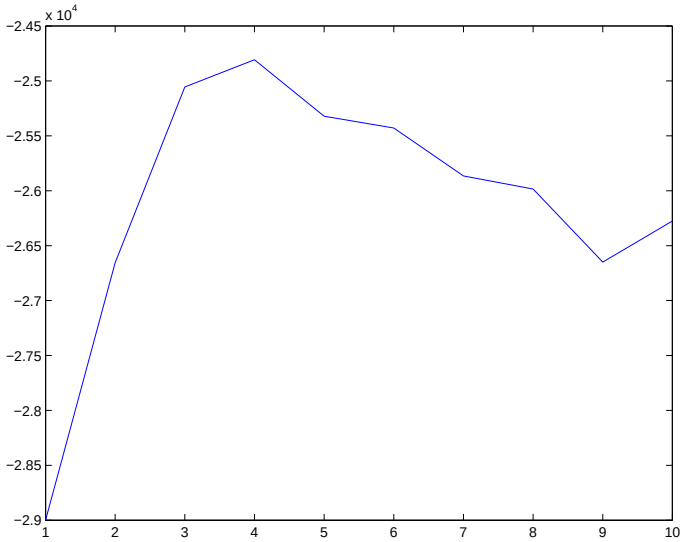


Fig. 6.32: Free energy $F(\mathbf{\Theta})$ (a bound lower on the evidence $p(Data|K)$) against K for the data set of Sorlie *et al*

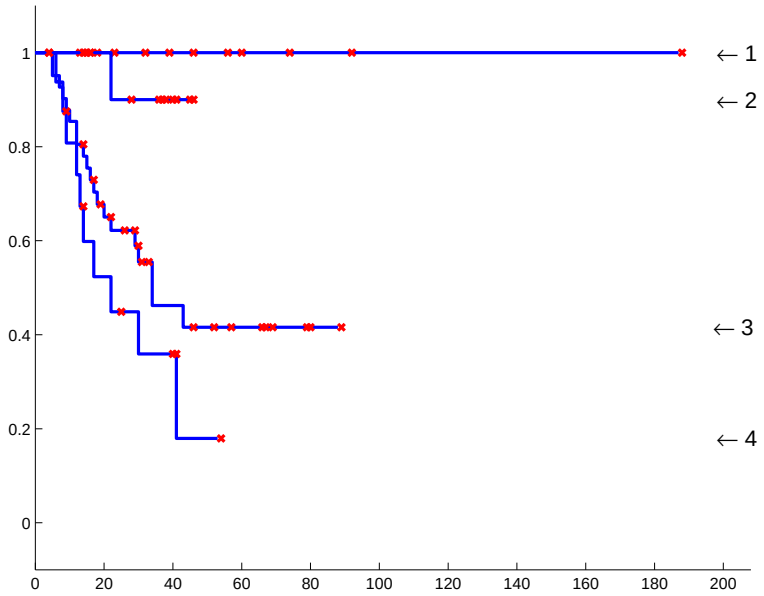


Fig. 6.33: Kaplan-Meier plots for the Sorlie *et al* dataset. The graphs show fraction not expired from the disease (y-axis) versus number of months (x-axis). There are 15 patients in process 1, 14 in 2, 41 in 3 and 17 in 4 (the remaining samples are insufficiently identified with a process). A vertical drop indicates expiry from the disease and a star indicates the patient is not recorded as expired from the disease (this includes the point at which some patients exited the survey).

6.7 Conclusion

The results from the four datasets used in this study are broadly consistent and indicate at least four principal processes for breast cancer. However, as illustrated with comparison to the prostate cancer data set studied in [70], the distinction between processes is less clear than with other cancers, reflecting the heterogeneous nature of this disease.

Our analysis suggests the existence of an indolent subtype distinguished by under-expression across a number of genes associated with tumour growth. There is a subtype closely related to the Luminal A subtype proposed by Sorlie *et al* [78]. In line with previous observations there is also a subtype marked by up-regulation of *ERBB2* (*HER2*) and *GRB7*. The most aggressive subtype is also the most well defined. This subtype is marked by abnormal expression of the transcription factor genes *FOXA1*, *FOXC1*, *GATA3*, *TFF3* and *XBP1*, for example, and it is associated with loss of regulation of the vascular growth factor *VEGF*. As already remarked, using a sparse graphical model [27, 28], we find that the transcription factor genes *FOXA1*, *GATA3*, *TFF3* and *XBP1* appear closely linked with the estrogen receptor-alpha gene, which with the estrogen pathway, plays a crucial role in the development of many breast tumours. One target of ERα is the *TFF1* gene and *FOXA1* has a direct influence on transcription by this gene since there are binding sites for *FOXA1* in its promoter region [10]. A number of other ERα-bound promoters have *FOXA1* binding sites [51]. The role of *FOXA1* has been highlighted in a contemporary study by Laganiere *et al* [51]: expression by *FOXA1* correlates with the presence of ERα and it has been suggested that that this gene plays a crucial role in a transcriptional domain governing estrogen response. Reinforcing this result, a contemporary study by Carroll *et al* [22] has shown that forkhead factor binding sites are present in 54% of 57 ER binding regions. This strongly supports the significance of abnormal expression of *FOXA1* and *FOXC1* indicated by our analysis. Finally, in agreement with the analysis using a sparse graphical model [27, 28], there appears to be an important role played by *TFF3*, a close relative of *TFF1*.

The decomposition proposed here is at most a basic model since one would expect further subdivision as more data becomes available, thus enabling a higher resolution picture. As remarked previously, the effects of noise are averaged out as the dataset size increases. Thus for the dataset of Sorlie *et al* the peak in the likelihood curve is at 3-4 processes but, for the

largest dataset of de Vijver *et al*, it is approximately 4-5. Certainly, our analysis suggests that the 2 process split of de Vijver *et al* [25] is too simple a model and at least 4 main processes are justified by the datasets used. The dataset for West *et al* was exclusively based on invasive ductal tumours and the Sorlie *et al* dataset had samples very predominantly of this type. However, use of samples consistently of the same histological type would also help reduce noise and improve definition. The indolent subtype 1 was not presented in the original analysis of Sorlie *et al* and the ability of the method to find this feature highlights the importance of using Probabilistic methods in this context.

6.8 Supplementary comment on the dataset of De Vijver *et al.*

In the original publication of de Vijver *et al* [25] 21 cDNA sequences had no gene name or information associated with them. Given this fact and the monotonic trends in mean expression values mentioned in the main text we have updated and examined ontology information for the 70 genes and their encoded proteins to examine their significance. A full description of all 70 entries and further information is available as supplementary data at www.enm.bris.ac.uk/lpd/bc.htm. In the table below we list the top ranked genes distinguishing process 1 vs process 4 (with $Z_1 > 2$) for the dataset of de Vijver *et al*. The 4 columns headed *Process* are the mean logged expression values (using log base 10). The processes are ranked in order of most indolent (1) to most aggressive (4) outcome. The end column highlights the progression trend across the 4 processes. Genes marked *BCSS1* and *BCSS2* correspond to hypothetical genes: *BCSS1* is 'moderately similar to T50635 hypothetical protein' and *BCSS2* is 'weakly similar to ISHUSS disulfide-isomerase'. The Z_1 values follow a normal probability distribution $\mathcal{N}(0, 1)$.

Of these genes, *ORC6L* is involved in DNA replication and serves as a platform for the assembly of additional initiation factors such as *CDC6* and *MCM*. siRNA gene silencing studies indicate that *ORC6L* plays an essential role in coordinating chromosome replication and segregation with cytokinesis. *STK32B* is a serine/threonine kinase. *KIAA1442* encodes a transcription factor with an IPT/TIG motif. These motifs are found in cell surface receptors such as Met and Ron as well as in intracellular transcription factors where it is involved in

DNA binding. Intriguingly the Ron tyrosine kinase receptor shares with the members of its subfamily (Met and Sea) the control of cell dissociation, motility, and invasion of extracellular matrices (scattering) [24]. Two genes have no known function though Contig38288RC is weakly similar to ISHUSS protein disulfide-isomerase, an enzyme that participates in the folding of proteins containing disulfide bonds. In the Table we have labelled Contig55725RC as *BCSS1* and Contig38288RC as *BCSS2* (breast cancer survival signature 1 and 2). Many genes are involved in processes associated with tumour growth such as DNA replication (*MCM6*), cell cycle control (*CCNE2*), spindle associated factors (*NUSAP1*, *PRC1*), chromosome organisation (*CENPA*), actin filament assembly (*DIAPH3*) and vascular remodelling (*ITS*). All these genes are up-regulated for the most aggressive process versus the least aggressive. *DIAPH3*, which was unidentified in the original paper, appears three times in the 70 gene set.

Gene ID	Gene name	Process 1	Process 2	Process 3	Process 4	Z_1	Trend
NM_014321	*ORC6L*	−0.47	−0.32	−0.02	0.26	4.29	Up
Contig55725_RC	*BCSS1*	−0.80	−0.54	−0.22	0.39	4.15	Up
NM_018401	*STK32B*	0.32	0.07	0.01	−0.11	3.14	Down
AB037863	*KIAA1442*	0.28	0.05	−0.01	−0.29	3.07	Down
Contig38288_RC	*BCSS2*	−0.34	−0.16	−0.02	0.26	3.06	Up
NM_003981	*PRC1*	−0.45	−0.30	0.02	0.24	2.98	Up
NM_016359	*NUSAP1*	−0.50	−0.28	0.039	0.22	2.93	Up
NM_004702	*CCNE2*	−0.55	−0.32	−0.02	0.22	2.93	Up
NM_001809	*CENPA*	−0.52	−0.41	−0.06	0.29	2.80	Up
AL137718	*DIAPH3*	−0.30	−0.10	0.03	0.22	2.78	Up
NM_014791	*MELK*	−0.46	−0.21	0.01	0.26	2.71	Up
NM_016448	*RAMP*	−0.36	−0.17	0.05	0.15	2.65	Up
Contig40831_RC	*AI224578*	−0.39	−0.11	−0.05	0.19	2.57	Up
AL080059	*TSPYL5*	−0.53	−0.24	−0.15	0.25	2.50	Up
Contig46218_RC	*DIAPH3*	−0.35	−0.22	0.04	0.27	2.50	Up
NM_003875	*GMPS*	−0.34	−0.17	−0.05	0.21	2.45	Up
NM_020974	*SCUBE2*	0.24	0.19	−0.24	−0.99	2.39	Down
NM_000436	*OXCT1*	−0.29	−0.06	−0.10	0.15	2.37	Mixed
NM_005915	*MCM6*	−0.37	−0.14	0.00	0.23	2.31	Up
AA555029_RC	*AA555029*	−0.31	−0.09	−0.06	0.15	2.27	Up
NM_002916	*RFC4*	−0.29	−0.133	−0.01	0.20	2.27	Up
AL080079	*GPR126*	−0.59	−0.25	−0.12	0.17	2.22	Up
NM_015984	*UCHL5*	−0.21	−0.08	−0.01	0.15	2.13	Up
Contig20217_RC	*TGS*	−0.33	−0.17	−0.02	0.17	2.08	Up
NM_006117	*PECI*	0.21	0.05	0.01	−0.25	2.07	Down
Contig32185_RC	*ITS*	−0.33	−0.14	−0.08	0.15	2.02	Up

Chapter 7

Conclusions

In this thesis we set out to apply some novel probabilistic models to biomedical data. The probabilistic approach to modelling data, and in particular the Bayesian framework, has been shown to be very successful in tackling a wide range of problems and will no doubt become more popular in the future. Probabilistic methods build on a huge bank of mathematical literature. With a wealth of well understood distributions and theory they are extremely flexible. Models are formulated in an explicit and analytical way, with a clear generative process. The graphical formulation introduced in chapter 2 provides a easy way to visualise the dependencies and relationships between variables in the models. One advantage that probabilistic approaches have over some alternatives is that since they are based on full distributions they can give an analytical measure of the confidence in the conclusions drawn. For example, in the case of an SVM the classification label is assigned as ± 1 depending on which side of the separating hyperplane the test point lies, but within this framework there is no analytical mechanism to determine the confidence we have in a classification. Probabilistic methods can also formally indicate the significance of a particular data point, eg expressions for a particular gene.

We will now give a short summary and comment on each chapter in turn and suggest possible extensions.

7.1 Chapter 3: A Hierarchical Representation of Lung Disease

Through a novel model, chapter 3 provided some interesting analysis of radiological data, and in addition demonstrated the use of variational inference. The motivation was to automatically learn a two-level hierarchical representation of lung disease. To construct the model we took the re-sampling based approach of the Latent Dirichlet Allocation setting of Blei *et al* [15] and extended it in two ways. Firstly we replaced the original *bag of words* multinomial assumption with a set of Gaussian distributions for image features. Secondly we added a second generative level that was linked to the first via a multinomial. This gave rise to a two-tier hierarchy in which an individual CT image would be decomposed in regions of similar texture. The results were hard to quantify absolutely as no validation data set was available.

There are many obvious extensions to the model. Any number of tiers can be added to the hierarchical structure, indicating a greater granularity in disease. However each new tier added would increase the number of parameters to be estimated and increase the complexity of the update equations. One drawback in the the model is that cross validated maximum likelihood analysis would only give rise to the same number of processes in the upper and lower levels of the hierarchy. A simple approach to remedy this, and one that does not involve changing the model, is to impose different prior distributions on the variances σ^2_{fk} of the Gaussian distributions. Prior distributions that favoured larger variances for upper level and smaller variances ones for the lower level would be a suitable choice. Finally an increase in the amount of data used would guarantee a more representative selection of the wide ranging appearances of chest CT images, and lead to a better overall representation of lung disease.

7.2 Chapter 4: Unsupervised Learning in Radiology

Two separate data sets consisting of textual radiology reports and corresponding CT images were jointly modelled in chapter 4. The purpose of this work was to see to what extent it

is possible to learn from radiological data in an entirely unsupervised manner. A variety of probabilistic models which jointly modelled the textual and image information were analysed. In likelihood comparisons, those with a re-sampling element based on LDA ([15]) we shown to be superior. Indeed it was shown that the methods used automatically highlighted subtypes of disease.

As always, a larger and more varied data set would have improved results. We were somewhat limited by the range of disease types available for the study. This was reflected in the limited vocabulary of *Emphysema*, *Fibrosis* and *Normal* that was used. A wider ranging vocabulary with a roughly even number of samples for each type would have extended the study. One simple extension to this work would be to include hand labelled data. At the moment the the algorithm is entirely unsupervised, but it is straightforward to introduce a supervised element. This is done by creating *faked* report-image pairings. These are such that the image is not a full CT scan, but made up of a labelled region. The corresponding report is the single label for that region. This *faked* data can be simply augmented with the original data.

7.3 Chapter 5: Joint Estimation of Motif and Gene Expression Data

In chapter 5 we attempted to model the relationship between upstream DNA motif abundance and gene expression in a yeast stress-test data set.

The models chosen were taken as variations on the correspondence LDA of Blei *et al* [16]. These models proved to be better in likelihood comparison than simpler LDA models, mixture based approaches and a null model. However, the predictive power of these models was shown to be relatively poor. Although this was a negative result there are a number of positive conclusions that we can draw from the detailed analysis. The expression data appeared to have much more structure, in terms of coherent groupings of data points, than the motif data. Indeed some sets of motifs used were extremely degenerate. This suggested that the increased likelihood shown by the more complex models could be solely down to more sophisticated modelling of the expression data rather than the data set as a whole. It does indicate that, at least on this dataset, one should be wary of prediction of expression directly

from motif data. The interactions between motifs are biologically extremely complex and so one would expect only a non-linear model would be able to provide adequate prediction. One possible extension would be to include more, and varied, data. In Middendorf *et al* [61] they incorporate expressions of know *parents* (transcription factors) into the mode. In this case these *parents* also appear as genes in the data set, which suggests a possible circular prediction.

Technical extensions to the method include using a more sophisticated method of inference. It would be possible to construct a Monte Carlo simulation to provide full posterior distributions for all model parameters. Due to the high dimensionality of the model and the data set this would perhaps be prohibitive for practical analysis. The variational approach provides only point estimates of parameters, but confident assertions about the overall performance of a model can be given based on these estimates. It is for this reason that a full Monte Carlo simulation would probably be of little benefit.

7.4 Chapter 6 Prognostic Signatures in Breast Cancer

Chapter 6 has probably provided the most interesting results of this thesis. In this chapter we took four independent gene expression data sets for breast carcinomas. As an initial investigation we wanted to analyse the data to establish if there existed natural grouping between the patients. We took the LPD algorithm of [70] and applied it to these data sets. As well as the variational approach to inference originally used by the authors we performed Monte Carlo simulations to confirm the findings. The results showed some striking trends in Breast cancer that were previously unknown. It was shown consistently that there were a greater number of distinct subtypes than the commonly assumed number of 2. *A priori* we assumed the number of subtypes to be unknown. A figure of 4 subtypes was suggested from the data sets analysed, but this may increase when larger studies become available. Moreover the subtypes seemed to be consistently defined, by abnormal gene expression, across the data sets. The most distinct subtype also corresponded to the the most aggressive. It is marked by abnormal expression of a number of genes, in particular *FOXA1*, *FOXC1*, *GATA3*, *TFF3* and *XBP1*. Some of these genes have appeared together in previous unrelated analyses of breast cancer.

The next stage in the analysis would be to apply LPD to more publicly available data sets, such as that used in [86]. This aim of this work would be give a further estimate for the number of subtypes of disease present in the data set. In addition, we would hope it confirms the very distinct set of genes that are closely connected to the aggressive subtype. More ambitious further work would be to fuse all available breast cancer data sets into one large data set. This was attempted in the study of Segal *et al* [72], for a range data sets covering different cancers. It is inherently a hard problem as each independent data set was generated in different circumstances and so it is not valid to trivially compare absolute gene expression values. Based on the promising results of the breast cancer study, an obvious future direction would be to apply similar analysis to gene expression data sets from other cancers.

Bibliography

[1] http://www.stat.umn.edu/ charlie/mcmc/burn.html.

[2] Source http://cats.med.uvm.edu/.

[3] Source http://www.affymetrix.com/.

[4] Source: http://www.lottery.co.uk/stats/.

[5] A.A. Alizadeh, M. Eisen, R. Davis, C. Ma, I. Lossos, A. Rosenwald, J. Boldrick, H. Sabet, T. Tran, Y. Xin, J. Powell, L. Yang, G. Marti, T. Moore, J. Hudson, L. Lu, D. Lewis, R. Tibshirani, G. Sherlock, W. Chan, T. Greiner, D. Weisenburger, J. Armitage, R. Warnke, R. Levy, W. Wilson, M. Grever, J. Byrd, D. Botstein, P. Brown, and L. Staudt. Different types of diffuse large b-cell lymphoma identified by gene expressing profiling. *Nature*, 403:503–511, 2000.

[6] C. Andrieu, N. de Freitas, A. Doucet, and M. Jordan. An introduction to mcmc for machine learning. *Machine Learning*, 50:5–43.

[7] H. Attias. A variational bayesian framework for graphical models, 2000.

[8] Kobus Barnard, Pinar Duygulu, David Forsyth, Nando de Freitas, David M. Blei, and Michael I. Jordan. Matching words and pictures. *J. Mach. Learn. Res.*, 3:1107–1135, 2003.

[9] Kobus Barnard and David Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, pages II:408–415, 2001.

[10] S Beck, P Sommer, E Do Santos Silva, N Blin, and P Gott. Hepatocyte nuclear factor 3 (winged helix domain) activates trefoil factor gene TFF1 through a binding motif adjacent to the TATA box. *Cell Biology*, 18:157–164, 1999.

[11] A. Ben-Hur, D. Horn, H.T. Siegelmann, and V. Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125–137, 2001.

[12] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[13] J. Bilmes. A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. 1997.

[14] Christopher M. Bishop. *Neural networks for pattern recognition*. Oxford University Press, Oxford, UK, UK, 1996.

[15] D Blei, A Ng, and M Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[16] David M. Blei and Michael I. Jordan. Modeling annotated data. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 127–134, New York, NY, USA, 2003. ACM Press.

[17] M. Brown, W. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. Furey, M. Jr, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using suport vector machines. In *Proc. Natl. Acad. Sci.*, volume 97, pages 262–267, 2000.

[18] J P Brunet, P Tamayo, T R Golub, and J P Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings National Academy Sciences*, 101:4164–4169, 2004.

[19] L Carrivick, S Prabhu, P Goddard, and J Rossiter. Unsupervised learning in radiology using novel latent variable models. In *CVPR (2)*, pages 854–859, San Diego, California, June 2005.

[20] L. Carrivick, S. Rogers, J. Clark, C. Campbell, M. Girolami, and C. Cooper. Identification of prognostic signatures in breast cancer microarray data using bayesian techniques. *Journal of the Royal Society Interface*, 2005.

[21] Luke Carrivick and Sanjay Prabhu. Deriving a hierarchical representation of lung disease using re-sampling mixture models. In *Medical Image Understanding and Analysis (MIUA)*, pages 155 – 158, Bristol, UK, July 2005.

[22] JS Carroll, XS Liu, AS Brodsky, W Li, CA Meyer, AJ Szary, J Eeckhoute, W Shao, EV Hestermann, TR Geistlinger, EA Fox, PA Silver, and M. Brown. Chromosome-wide mapping of estrogen receptor binding reveals long-range regulation requiring the forkhead protein FOXA1. *Cell*, 122:33–43, 2005.

[23] Gilles Celeux, Stéphane Chrétien, Florence Forbes, and Abdallah Mkhadri. A component-wise EM algorithm for mixtures. *Journal of Computational and Graphical Statistics*, 10(4):697–717, 2001.

[24] C Collesi, M Santoro, G Gaudino, and P Comoglio. A splicing variant of the RON transcript induces constitutive tyrosine kinase activity and an invasive phenotype. *Molecular Cellular Biology*, 16:5518–5526, 1996.

[25] M de Vijver et al. A gene expression signature as a predictor of survival in breast cencer. *New England Journal of Medicine*, 347:1999–2009, 2002.

[26] A.P. Dempster, N. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[27] A Dobra, B Jones, C Hans, J Nevins, and M West. Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis*, 90:196–212, 2004.

[28] A Dobra and M West. Graphical model-based gene clustering and metagene expression analysis. Technical report, 2004.

[29] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* Wiley-Interscience Publication, 2000.

[30] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci USA*, 95:14863–14868, December 1998.

[31] M Elkin, A Orgel, and H Kleinman. An angiogenic switch in breast cancer involves estrogen and soluble vascular endothelial growth factor receptor 1. *Journal of the National Cancer Institute*, 96:875–978, 2004.

[32] Chun-Hua Wang *et al.* Persistence of lung inflammation and lung cytokines with high-resolution ct abnormalities during recovery from sars. *Respiratory Research*, 6, 2005.

[33] M. Brown *et al.* Method for segmenting chest ct image data using an anatomical model: preliminary results. *IEEE Trans. Med. Imaging*, 16(6):828–838, 1997.

[34] P Flaherty, G Giaever, J Kumm, M I Jordan, and A P Arkin. A latent variable model for chemogenomic profiling. *Bioinformatics*, 21:3286–3293, 2005.

[35] Caroll Friedman. Towards a comprehensive medical language processing system: methods and issues. *Proc AMIA Annu Fall Symp*, 1:595–599, 1997.

[36] A P Gasch et al. Genomic expression program in the response of yeast cells to environmental changes. *Molecular Biology of the Cell*, 11:4241–4257, 2000.

[37] Mark Girolami and Simon Rogers. Hierarchic bayesian models for kernel learning. In *ICML: 22nd International Conference on Machine Learning*, Bonn, Germany., August 2005.

[38] Paul R Goddard. *Diagnostic Imaging of the Chest.* Churchill Livingstone, 1987.

[39] T. Griffiths and M. Steyvers. Finding scientific topics. In *Proceedings of the National Academy of Sciences*, volume 101, pages 5228–5235, 2004.

[40] S Gruvberger et al. Estrogen receptor status in breast cancer is associated with remarkably dstinct gene expression patterns. *Cancer Research*, 61:5979–5984, 2001.

[41] I Hedenfalk et al. Gene-expression profiles in hereditory breast cancer. *New England Journal of Medicine*, 344:539–548, 2001.

[42] Ralf Herbrich, Thore Graepel, and Colin Campbell. Bayes point machines. *Journal of Machine Learning Research*, 1:245–279, 2001.

[43] E. J. Horvitz, J. S. Breese, and M. Henrion. Decision theory in expert systems and artificial intelligence. *International Journal of Approximate Reasoning*, 2:247–302, 1988.

[44] Shiying Hu, Eric A. Hoffman, and Joseph M. Reinhardt. Automatic lung segmentation for accurate quantitation of volumetric x-ray ct images. *IEEE Trans. Med. Imaging*, 20(6):490–498, June 2001.

[45] J. L. W. V. Jensen. Sur les functions convexes et les ingalits entre les valeurs moyennes. *Acta Mathematica*, 30:175–193, 1906.

[46] David Delany Jonathan Corne, Mary Carroll and Ivan Brown. *Chest x-ray made easy.*

[47] F Katsuoka, H Motohashi, J Engel, and M Yamamoto. NRF2 transcriptionally activates the MAFG gene through an antioxidant response element. *J Biol Chem*, 280:4483–4490, 2005.

[48] Risi Imre Kondor and John D. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning*, pages 315–322, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[49] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[50] M Lacroix and G Leclerq. About GATA3, HNF3A and XBP1, three genes co-expressed with the oestrogen receptor-alpha gene (ESR1) in breast cancer. *Molecular and Cellular Endocrinology*, 219:1–7, 2004.

[51] J et al Laganiere. Location analysis of estrogen receptor α target promoters reveals that FOXA1 defines a domain of the estrogen response. *Proceedings National Academy Sciences*, 102:11651–11656, 2005.

[52] G R G Lanckriet, T De Bie, N Cristianini, M I Jordan, and W S Noble. A statistical framework for genomic data fusion. *Bioinformatics*, 20:2626–2635, 2004.

[53] Gert R. G. Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I. Jordan. Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.*, 5:27–72, 2004.

[54] Pat Langley, Wayne Iba, and Kevin Thompson. An analysis of bayesian classifiers. In *National Conference on Artificial Intelligence*, pages 223–228, 1992.

[55] Y. Li, C. Campbell, and M. Tipping. Bayesian automatic relevance determination algorithms for classifying gene expression data. *Bioinformatics*, 18:1332–1339, 2002.

[56] D. J. C. MacKay. Introduction to Monte Carlo methods. In M. I. Jordan, editor, *Learning in Graphical Models*, NATO Science Series, pages 175–204. Kluwer Academic Press, 1998.

[57] David J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, 2002.

[58] G McLachlan, R Bean, and D Peel. A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18(3):413–422, 2002.

[59] G McLachlan and D Peel. *Finite Mixture Models*. John Wiley Inc, 2000.

[60] N. Metropolis, A. Rosenbluth, M. Rosenbluth, M. Teller, and E. Teller. Equations of state calculations by fast computing machines. *J. Chem. Phys.*, 21, 1953.

[61] M Middendorf, A Kundaje, C Wiggins, Y Freund, and C Leslie. Predicting genetic regulatory response using classification. In *Proceedings of the Twelfth International Conference on Intelligent Systems in Molecular Biology (ISMB 2004)*, page in press, 2004.

[62] Thomas P. Minka. *A family of algorithms for approximate bayesian inference*. PhD thesis, 2001. Supervisor-Rosalind Picard.

[63] T D Moloshok, R R Klevecz, J D Grant, F J Manion, W F Speier, and M F Ochs. Application of bayesian decomposition for analysing microarray data. *Bioinformatics*, 18:566–575, 2002.

[64] Aljaz Noe and James C. Gee. Partial volume segmentation of cerebral mri scans with mixture model clustering. In *IPMI*, pages 423–430, 2001.

[65] A. Jeffries P. McGowan and A. Turley. *Crash Course. Respiratory System.*

[66] W.D. Penny and S.J. Roberts. Variational bayes for 1-dimensional mixture models. Technical report, Department of Engineering Science, Oxford University, 2000.

[67] S Pero, R Daly, and D Krag. GRB7-based molecular therapeutics in cancer. *Expert Reviews in Molecular Medicine*, 5:1–11, 2003.

[68] C Perou et al. Molecular portraits of human breast tumours. *Nature*, 406:747–752, 2000.

[69] D Rees. *Essential Statistics*. Chapman and Hall, 2001.

[70] S Rogers, M Girolami, C Campbell, and R Breitling. The latent process decomposition of cDNA microarray datasets. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2, 2005 (to appear).

[71] E Segal, A Battle, and D Koller. Decomposing gene expression into cellular processes. In *Proc. 8th Pacific Symposium on Biocomputing (PSB)*, pages 89–100, 2003.

[72] Eran Segal, Nir Friedman Daphne Koller, and Aviv Regev. A module map showing conditional activity of expression modules in cancer. *Nature Genetics*, (36):1090–1098, september 2004.

[73] Eran Segal, R. Yelensky, and Daphne Koller. Genome-wide discovery of transcriptional modules from dna sequence and gene expression. In *ISMB (Supplement of Bioinformatics)*, pages 273–282, 2003.

[74] J Shawe-Taylor and N Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[75] E. Shortliffe. *Computer-based Medical Consultations: MYCIN.*

[76] I.C. Sluimer, P.F. van Waes, M.A. Viergever, and B. van Ginneken. Computer-aided diagnosis in high-resolution CT of the lungs. *Medical Physics*, 30(12):3081–3090, 2003.

[77] T Sorlie et al. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings National Academy Sciences*, 98:10869–10874, 2001.

[78] T Sorlie et al. Repeated observation of breast tumor subtypes in independent gene expression data sets. *Proceedings National Academy Sciences*, 100:8418–8423, 2003.

[79] Soderland SG Taira RK. A statistical natural language processor for medical reports.

[80] Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[81] R. Uppaluri et al. Computer recognition of regional lung disease patterns. *Am. J. Respir. Crit. Care Med*, 160:648–654, 1999.

[82] R. Uppaluri, E.A. Hoffman, M. Sonka, G.W. Hunninghake, and G. McLennan. Interstitial lung disease . a quantitative study using the adaptive multiple feature method. *Am. J. Respir. Crit. Care Med*, 159(2):519–525, 1999.

[83] L van 't Veer et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–535, 2002.

[84] Vladimir N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[85] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. Technical Report 649, Department of Statistics, University of California, Berkeley, 2003., 2003.

[86] Y Wang, J Klijn, Y Zhang, et al. Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet*, pages 671–679, 2005.

[87] M West et al. Predicting the clinical status of human breast cancer using gene expression profiles. *Proceedings of the National Academy of Sciences*, 98:11462–11467, 2001.

[88] E Wingender and et al. TRANSFAC: an integrated system for gene expression regulation. *Nucleic Acids Research*, 28:316–319, 2000.

[89] Hastings W.K. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57, 1970.

[90] L Zhukova, N Zhukov, and M Lichinitser. Expression of FLT-1 and FLK-1 receptors for vascular endothelial growth factor on tumor cells as a new prognostic criterion for locally advanced breast cancer. *Bull Exp Biol Med*, 135:478–481, 2003.

Appendix A

Details of the LPD Gibbs Sampler

A.1 LDA Gibbs

Here we give a full derivation for two conditional distributions first stated in section 2.4.3 and subsequently used in chapter 6.

We wish to find an expression for the conditional distribution of the model parameter μ

$$P(\mu_{gk}|\alpha, \mu_{-(gk)}, \sigma, \theta, Z, E) = P(\mu_{gk}|\sigma_{gk}, \theta_d, Z_{.g}, E_{.g})$$

From joint distribution given in equation 2.96

$$P(\mu_{gk}|\sigma_{gk}, \theta_{.k}, Z_{.g}, E_{.g}) \propto \prod_d \left(\theta_{dk} \frac{1}{\sigma_{gk}} \exp(-\frac{(E_{dg} - \mu_{gk})^2}{2\sigma_{gk}^2}) \right)^{I_{\{Z_{dg}=k\}}} \frac{1}{\tau} \exp(-\frac{\mu_{gk}^2}{2\tau^2})$$

Thus by combining the prior and likelihood:

$$
\begin{aligned}
P(\mu_{gk}|\sigma_{gk},\theta_{.k},Z_{.g},E_{.g}) &\propto \prod_t \left(\theta_{tk}\tfrac{1}{\sigma_{gk}}\exp(-\tfrac{(E_{tg}-\mu_{gk})^2}{2\sigma_{gk}^2})\right)\tfrac{1}{\tau}\exp(-\tfrac{\mu_{gk}^2}{2\tau^2}) \\
&\propto \prod_t \left(\exp(-\tfrac{(E_{tg}-\mu_{gk})^2}{2\sigma_{gk}^2})\right)\exp(-\tfrac{\mu_{gk}^2}{2\tau^2}) \\
&= \left(\exp(-\tfrac{1}{2\sigma_{gk}^2\tau^2}(\textstyle\sum_t(\tau^2E_{tg}^2-2\tau^2\mu_{gk}E_{tg}+\tau^2\mu_{gk}^2)+\sigma_{gk}^2\mu_{gk}^2))\right) \\
&\propto \left(\exp(-\tfrac{1}{2\sigma_{gk}^2\tau^2}(\textstyle\sum_t(-2\tau^2\mu_{gk}E_{tg}+\tau^2\mu_{gk}^2)+\sigma_{gk}^2\mu_{gk}^2))\right) \\
&= \exp(-\tfrac{1}{2\sigma_{gk}^2\tau^2}(\mu_{gk}^2(T\tau^2+\sigma_{gk}^2)-2\tau^2\mu_{gk}\textstyle\sum_t E_{tg})) \\
&= \exp(-\tfrac{T\tau^2+\sigma_{gk}^2}{2\sigma_{gk}^2\tau^2}(\mu_{gk}^2-\tfrac{2\tau^2\mu_{gk}\sum_t E_{tg}}{T\tau^2+\sigma_{gk}^2})) \\
&\propto \exp(-\tfrac{T\tau^2+\sigma_{gk}^2}{2\sigma_{gk}^2\tau^2}(\mu_{gk}-\tfrac{\tau^2\sum_t E_{tg}}{T\tau^2+\sigma_{gk}^2})^2) \\
&\sim \mathcal{N}(\tfrac{\tau^2\sum_t E_{tg}}{T\tau^2+\sigma_{gk}^2},\tfrac{\sigma_{gk}^2\tau^2}{T\tau^2+\sigma_{gk}^2}) \\
&= \mathcal{N}(\tfrac{\tau^2\sum_t E_{tg}}{T\tau^2+\sigma_{gk}^2},(\tfrac{T}{\sigma_{gk}^2}+\tfrac{1}{\tau^2})^{-1})
\end{aligned}
\tag{A.1}
$$

the posterior for μ is a Gaussian distribution $\mathcal{N}\left(\frac{\tau^2\sum_t E_{tg}}{T\tau^2+\sigma_{gk}^2},\left(\frac{T}{\sigma_{gk}^2}+\frac{1}{\tau^2}\right)^{-1}\right)$. Similarly for the parameter σ^2

$$
\begin{aligned}
P(\sigma_{gk}|\mu_{gk},\theta_{.k},Z_{.g},E_{.g}) &\propto \prod_t \left(\tfrac{1}{\sigma_{gk}}\exp(-\tfrac{(E_{tg}-\mu_{gk})^2}{2\sigma_{gk}^2})\right)\sigma_{gk}^{-s} \\
&= \sigma^{-(s+T)}\exp(-\tfrac{\sum_t(E_{tg}-\mu_{gk})^2}{2\sigma_{gk}^2}) \\
&\sim InverseGamma((s+T)/2,\tfrac{1}{2}\textstyle\sum_t(E_{tg}-\mu_{gk})^2)
\end{aligned}
\tag{A.2}
$$

Appendix B

Derivation of a Hierarchical Mixture Model

B.1 Hierarchical Extension

We will give a full derivation of the variational EM algorithm use for inference in the model of chapter 3

B.1.1 Derivation of update equations

Starting with the likelihood, and writing $\prod_f P(R_{ndf}) = P(R_{nd})$

$$\begin{aligned} P(R_d|\mu,\sigma,\beta,\alpha) = & \int_{\triangle} \prod_n^{N_d} \sum_k P(R_{nd}|Z_n=k,\mu,\sigma)P(Z_n=k|\theta)P(\theta|\alpha) \\ & \prod_n \sum_{k,k'} P(R_{nd}|Y_n=k',\mu^{'},\sigma^{'})P(Y_n=k'|Z_n=k)P(Z_n=k|R_{nd},\theta)d\theta \\ & = \int_{\triangle} \zeta_1\zeta_2 d\theta \end{aligned} \tag{B.1}$$

Where

$$\zeta_1 = \prod_n \sum_k P(R_{nd}|Z_n = k, \mu, \sigma) P(Z_n = k|\theta) P(\theta|\alpha)$$

and

$$\zeta_2 = \prod_n \sum_{k,k'} P(R_{nd}|Y_n = k', \mu^{'}, \sigma^{'}) P(Y_n = k'|Z_n = k) P(Z_n = k|R_{nd}, \theta)$$

Consider the full data log-likelihood $\sum_g \log P(R_d|\mu, \sigma, \beta, \alpha)$. Via the introduction of two variational distributions: a sample specific Dirichlet distribution with parameters γ_{dk} and a sample and region specific multinomial with parameter ϕ_{ndk}, we can use Jensen's inequality twice and create a bound on ζ_1.

$$\int_\triangle \sum_d \log(\zeta_1) d\theta \geq \int_\triangle \sum_{d,n,f,k} P(\theta|\gamma_d) \phi_{ndk} \log \left[\frac{P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk}) \theta_k}{P(\theta|\gamma_d) \phi_{ndk}} P(\theta|\alpha) \right] d\theta \quad \text{(B.2)}$$

Secondly using Jensen's inequality for the expectation of $P(Z_n = k|R_{nd}, \theta) = \phi_{ndk}$ and introducing a discrete variational distribution with parameters $\eta_{ndk'}$ we can give a bound on ζ_2.

$$\begin{aligned}
\textstyle\sum_d \log(\zeta_2) = \quad & \textstyle\prod_{n,k} \sum_{k'} P(R_{nd}|Y_n = k', \mu', \sigma')\beta_{kk'}\phi_{ndk} \\
& = \textstyle\sum_{d,n} \log\left[\sum_{k,k'} P(R_{nd}|Y_n = k', \mu', \sigma')\beta_{kk'}\phi_{ndk}\right] \\
& \geq \textstyle\sum_{d,n,k} \phi_{ndk} \log\left[\sum_{k'} P(R_{nd}|Y_n = k', \mu', \sigma')\beta_{kk'}\right] \\
& = \textstyle\sum_{d,n,k} \phi_{ndk} \log\left[\sum_{k'} P(R_{nd}|Y_n = k', \mu', \sigma')\beta_{kk'}\frac{\eta_{ndk'}}{\eta_{ndk'}}\right] \\
& \geq \textstyle\sum_{d,n,k,k'} \phi_{ndk}\eta_{ndk'} \log\left[\frac{P(R_{nd}|Y_n=k', \mu', \sigma')\beta_{kk'})}{\eta_{ndk'}}\right]
\end{aligned} \tag{B.3}$$

We have written $P(Y_n = k'|Z_n = k) = \beta_{kk'}$. In total we now have:

$$\begin{aligned}
\textstyle\sum_d \log P(R_d|\mu, \sigma, \beta, \alpha) \quad & \geq \textstyle\int_\triangle \sum_{d,n,k} P(\theta|\gamma_d)\phi_{ndk} \log\left[\frac{P(R_{nd}|Z_n=k, \mu_{fk}, \sigma_{fk})\theta_k}{P(\theta|\gamma_d)\phi_{ndk}} P(\theta|\alpha)\right] d\theta \\
& + \textstyle\sum_{d,n,k,k'} \phi_{ndk}\eta_{ndk'} \log\left[\frac{P(R_{nd}|Y_n=k', \mu', \sigma')\beta_{kk'}}{\eta_{ndk'}}\right]
\end{aligned} \tag{B.4}$$

Expanding the logs, and using the summations $\sum_{k'} \eta_{ndk'} = 1$ and $\sum_k \phi_{ndk} = 1$ we have:

$$
\begin{aligned}
\log \mathcal{L} \quad &\geq \int_{\triangle} \sum_{d,n,f,k} P(\theta|\gamma_d)\phi_{ndk} \log P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk}) d\theta \\
&+ \int_{\triangle} \sum_{d,n,k} P(\theta|\gamma_d)\phi_{ndk} \log(\theta_k) d\theta \\
&+ \int_{\triangle} \sum_{d,n} P(\theta|\gamma_d) \log P(\theta|\alpha) d\theta \\
&- \int_{\triangle} \sum_{d,n} P(\theta|\gamma_d) \log P(\theta|\gamma_d) d\theta \\
&- \int_{\triangle} \sum_{d,n,k} P(\theta|\gamma_d)\phi_{ndk} \log(\phi_{ndk}) d\theta \\
&+ \sum_{d,n,f,k'} \eta_{ndk'} \log \left[P(R_{ndf}|Y_n = k', \mu', \sigma') \right] \\
&+ \sum_{d,n,k,k'} \phi_{ndk}\eta_{ndk'} \log \left[\beta_{kk'}\right] \\
&- \sum_{d,n,k'} \eta_{ndk'} \log \left[\eta_{ndk'}\right]
\end{aligned}
\tag{B.5}
$$

Derivation for μ, σ, α and γ

Taking the derivative of equation (B.5) with respect to μ_{fk}, including the only terms in μ and using

$$
P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk}) \sim \mathcal{N}(\mu_{fk}, \sigma_{fk}) \tag{B.6}
$$

we have

$$
\begin{aligned}
\frac{\partial}{\partial \mu_{fk}} \left(\sum_{d,n,f,k} \phi_{ndk} \log \left[\frac{P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk})\theta_k}{\phi_{ndk}} \right] \right) &= 0 \\
\Rightarrow \sum_{d,n} \frac{\phi_{ndk}(R_{ndf} - \mu_{fk})}{\sigma_{fk}^2} &= 0 \\
\mu_{fk} = \frac{\sum_{d,n} \phi_{ndk} R_{ndf}}{\sum_{d'} \phi_{nd'k}} &
\end{aligned}
\tag{B.7}
$$

Taking the derivative of equation (B.5) with respect to σ_{fk}, including the only terms in σ

we have

$$\frac{\partial}{\partial \sigma_{fk}} \left(\sum_{d,n,f,k} \phi_{ndk} \log \left[\frac{P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk})\theta_k}{\phi_{ndk}} \right] \right) = 0$$

$$\Rightarrow \sum_{d,n} \left(\frac{-\phi_{ndk}}{\sigma_{fk}} + \frac{\phi_{ndk}(R_{ndf} - \mu_{fk})^2}{\sigma_{fk}^3} \right) \tag{B.8}$$

$$\sigma_{fk}^2 = \frac{\sum_{d,n} \phi_{ndk}(R_{ndf} - \mu_{fk})^2}{\sum_{d'} \phi_{nd'k}} \tag{B.9}$$

Throughout the following we shall use the definition for a digamma function $\Psi = \frac{d}{dz}\log(\Gamma(Z))$, in terms of the Gamma function. We shall also use a consequence of general result for sufficient statistics that (see [**?**])

$$E[\log(\theta_i|\alpha)] = \int (\theta|\alpha)\log(\theta_i)d\theta = \left[\Psi(\alpha_i) - \Psi(\sum_j \alpha_j) \right] \tag{B.10}$$

By evaluating the integrals in the likelihood we have:

$$\begin{aligned} P(R_d|\mu, \sigma, \beta, \alpha) \quad & \geq \textstyle\sum_{d,n,k} \phi_{ndk} \log \left[P(R_{nd}|Z_n = k, \mu_{fk}, \sigma_{fk}) \right] \\ & + \textstyle\sum_{d,n,k} \phi_{ndk} \left[\Psi(\gamma_{dk}) - \Psi(\sum_k \gamma_{dk}) \right] \\ & + \textstyle\int_{\triangle} \sum_{d,n} P(\theta|\gamma_d) \log \left[P(\theta|\alpha) \right] d\theta \\ & - \textstyle\sum_{d,n} \phi_{ndk} \log \left[\phi_{ndk} \right] \\ & - \textstyle\int_{\triangle} \sum_{d,n} P(\theta|\gamma_d) \log \left[P(\theta|\gamma_d) \right] d\theta \\ & + \textstyle\sum_{d,m,k,n} \lambda_{nmd} \phi_{ndk} \log \left[P(W_m|Y_m = n, Z_n = k, \beta_{mk}) \right] \\ & - \textstyle\sum_{d,m,k,n} \lambda_{nmd} \log \left[\lambda_{nmd} \right] \end{aligned} \tag{B.11}$$

Evaluating the third term, we see that from the definition of a Dirichlet:

$$\log\left[P(\theta|\alpha)\right] = \log\left[\frac{\Gamma(\sum_k \alpha_k)}{\prod_k(\Gamma(\alpha_k))}\prod_k \theta_k^{\alpha_k-1}\right]$$

$$\begin{aligned}\int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\log\left[P(\theta|\alpha)\right]d\theta &= \int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\log\left[\Gamma(\sum_k \alpha_k)\right]d\theta \\ &- \int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\log\left[\prod_k(\Gamma(\alpha_k))\right]d\theta \\ &+ \int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\sum_k(\alpha_k-1)\log\theta_k d\theta\end{aligned} \tag{B.12}$$

$$= \sum_d^D \log\Gamma(\sum_k \alpha_k) - \sum_k \log\left[\Gamma(\alpha_k)\right] + \sum_{d,k}(\alpha_k-1)\left[\Psi(\gamma_{dk}) - \Psi(\sum_{k'}\gamma_{dk'})\right] \tag{B.13}$$

Evaluating the fifth term

$$\begin{aligned}\int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\log\left[P(\theta|\gamma_d)\right]d\theta &= \int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\log\left[\Gamma(\sum_k \gamma_{dk})\right]d\theta \\ &- \int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\log\left[\prod_k(\Gamma(\gamma_{dk}))\right]d\theta \\ &+ \int_\triangle \textstyle\sum_{d,n} P(\theta|\gamma_d)\sum_k(\gamma_{dk}-1)\log\theta_k d\theta\end{aligned} \tag{B.14}$$

$$= \log\Gamma(\sum_k \gamma_{dk}) - \sum_k \log\left[\Gamma(\gamma_{dk})\right] + \sum_{dk}(\gamma_{dk}-1)\left[\Psi(\gamma_{dk}) - \Psi(\sum_{k'}\gamma_{dk'})\right] \tag{B.15}$$

The only likelihood terms in α appear in equation (B.13). Taking the derivative with respect to α_i we have:

$$\sum_d\left[\Psi(\sum_k \alpha_k) - \Psi(\alpha_i) + \Psi(\gamma_{di}) - \Psi(\sum_{k'}\gamma_{dk'})\right] = 0 \tag{B.16}$$

It is impossible to find a first order closed form for an update of α_i, so an appropriate second order method must be used. The Newton-Raphson method used the second order Taylor series approximation for a function and has the closed form iterative update:

$$X_{n+1} = X_n - H_n^{-1}.g_n \tag{B.17}$$

for H the Hessian matrix, and g the gradient vector. In the case of α the gradient and Hessian are found by taking the first and second derivatives of the likelihood:

$$g(\alpha) = \frac{\partial L}{\partial \alpha_i} = D\left[\Psi(\sum_k \alpha_k) - \Psi(\alpha_i)\right] + \sum_d \left[\Psi(\gamma_{di}) - \Psi(\sum_{k'} \gamma_{dk'})\right] \tag{B.18}$$

$$H(\alpha) = \frac{\partial L^2}{\partial \alpha_i \partial \alpha_j} = D\Psi^{'}\left(\sum_k \alpha_k\right) - D\Psi^{'}(\alpha_i)\delta_{ij} \tag{B.19}$$

Where $D = \sum_d$, or the number of samples in the data.

Taking only those terms in γ:

$$\begin{aligned}
&\tfrac{\partial}{\partial \gamma_{dk}} P(R_d|\mu,\sigma,\beta,\alpha) \\
&= \tfrac{\partial}{\partial \gamma_{dk}} \textstyle\sum_{d,n,k} \phi_{ndk} \left[\Psi(\gamma_{dk}) - \Psi(\sum_{k'} \gamma_{dk'})\right] \\
&+ \tfrac{\partial}{\partial \gamma_{dk}} \textstyle\sum_{d,k} (\alpha_k - 1) \left[\Psi(\gamma_{dk}) - \Psi(\sum_{k'} \gamma_{dk'})\right] \\
&- \tfrac{\partial}{\partial \gamma_{dk}} \{\log \Gamma(\textstyle\sum_k \gamma_{dk}) - \sum_k \log\left[\Gamma(\gamma_{dk})\right]\} \\
&+ \tfrac{\partial}{\partial \gamma_{dk}} \{\textstyle\sum_{d,k} (\gamma_{dk} - 1) \left[\Psi(\gamma_{dk}) - \Psi(\sum_{k'} \gamma_{dk'})\right]\} \\
&= \tfrac{\partial}{\partial \gamma_{dk}} \textstyle\sum_{d,k} (\Psi(\gamma_{dk}) - \Psi(\sum_{k'} \gamma_{dk'}))(\sum_n \phi_{ndk} + \alpha_k - \gamma_{dk}) \\
&- \tfrac{\partial}{\partial \gamma_{dk}} \{\log \Gamma(\textstyle\sum_k \gamma_{dk}) - \sum_k \log\left[\Gamma(\gamma_{dk})\right]\} \\
&= -(\Psi(\gamma_{dk}) - \Psi(\textstyle\sum_{k'} \gamma_{dk'}) \\
&+ (\Psi^{'}(\gamma_{dk}) - \Psi^{'})(\textstyle\sum_{k'} \gamma_{dk'})(\sum_n \phi_{ndk} + \alpha_k - \gamma_{dk}) \\
&- (\Psi(\textstyle\sum_{k'} \gamma_{dk'} - \Psi(\gamma_{dk})) \\
&= (\Psi^{'}(\gamma_{dk}) - \Psi^{'})(\textstyle\sum_{k'} \gamma_{dk'})(\sum_n \phi_{ndk} + \alpha_k - \gamma_{dk}) \\
&= 0
\end{aligned} \tag{B.20}$$

Therefore, $\Psi^{'}(\gamma_{dk}) - \Psi^{'}(\sum_{k'} \gamma_{dk'}) = 0$ or $\sum_n \phi_{ndk} + \alpha_k - \gamma_{dk} = 0$. The γ_{dk} are all strictly greater than zero, so $\sum_{k'} \gamma_{dk'} \geq \gamma_{dk}$ and as the trigamma function $\Psi^{'}(x)$ is strictly monotonic decreasing for real $x \geq 0$ the first of the two possibilities cannot be zero. So the remaining solution, and hence update is:

$$\gamma_{dk} = \alpha_k + \sum_n \phi_{ndk} \tag{B.21}$$

Derivation for ϕ

Maximise equation (B.5) by taking the derivative with respect to ϕ_{ndk}, subject to the constraint $\sum_k \phi_{ndk} = 1$. We shall also use $\int P(\theta|\gamma_{dk})d\theta = 1$ and the identity from equation (B.10).

$$\begin{aligned}
&\textstyle\sum_f \log(P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk})) \\
&+ \textstyle\int_\triangle P(\theta|\gamma_d) \log(\theta_k) d\theta \\
&- P(\theta|\gamma_d) \left[\log(\phi_{ndk}) + 1\right] \\
&+ \textstyle\sum_{f,k'} \eta_{ndk'} \log \left[P(R_{ndf}|Y_n = k', \mu', \sigma')\right] \\
&+ \textstyle\sum_{k'} \eta_{ndk'} \log \left[\beta_{kk'}\right] \\
&- \textstyle\sum_{k'} \eta_{ndk'} \log \left[\eta_{ndk'}\right] \\
&= 0
\end{aligned} \tag{B.22}$$

$$\begin{aligned}
\Rightarrow \phi_{ndk} \quad &\propto \exp \left[\textstyle\sum_f \log \left[P(R_{ndf}|Z_n = k, \mu_{fk}, \sigma_{fk})\right]\right] \\
&\times \exp \left[\Psi(\gamma_{dk}) - \Psi(\textstyle\sum_k \gamma_{dk})\right] \\
&\times \exp \textstyle\sum_{f,k'} \eta_{ndk'} \log \left[P(R_{ndf}|Y_n = k', \mu', \sigma')\right] \\
&\times \exp \textstyle\sum_{k'} \eta_{ndk'} \left[\log \beta_{kk'} + \log \eta_{ndk'}\right]
\end{aligned} \tag{B.23}$$

Derivation for $\mu^{'}$ and $\sigma^{'}$

Taking the derivative of equation (B.5) with respect to $\mu^{'}_{fk'}$, including the only terms in μ and using

$$P(R_{ndf}|Y_n = k', \mu^{'}_{fk'}, \sigma^{'}_{fk'}) \sim \mathcal{N}(\mu^{'}_{fk'}, \sigma^{'}_{fk'}) = \frac{1}{(2\pi)^{\frac{1}{2}}\sigma^{'}_{fk'}} \exp\left[\frac{-(R_{ndf} - \mu^{'}_{fk'})^2}{2\sigma^{'2}_{fk'}}\right]$$

we have

$$\frac{\partial}{\partial \mu^{'}_{fk'}} \left(\sum_{d,n,f,k'} \eta_{ndk'} \log\left[P(R_{ndf}|Y_n = k', \mu^{'}_{fk'}, \sigma^{'}_{fk'})\right] \right) = 0$$

$$\Rightarrow \sum_{d,n} \eta_{ndk'}(R_{ndf} - \mu^{'}_{fk'}) = 0$$

$$\mu^{'}_{fk'} = \frac{\sum_{d,n} \eta_{ndk'} R_{ndf}}{\sum_{d'} \eta_{nd'k'}} \tag{B.24}$$

Taking the derivative of equation (B.5) with respect to $\sigma^{'}_{fk}$, including the only terms in $\sigma^{'}$ we have

$$\frac{\partial}{\partial \sigma^{'}_{fk'}} \left(\sum_{d,n,f,k'} \eta_{ndk'} \log\left[P(R_{ndf}|Y_n = k', \mu_{fk'}, \sigma^{'}_{fk'})\right] \right) = 0$$

$$\Rightarrow \sum_{d,n} \left(\frac{-\eta_{ndk'}}{\sigma^{'}_{fk'}} - \frac{\eta_{ndk'}(R_{ndf} - \mu_{fk'})^2}{\sigma^{'3}_{fk'}} \right)$$

$$\sigma^{'2}_{fk'} = \frac{\sum_{d,n} \eta_{ndk'}(R_{ndf} - \mu_{fk'})^2}{\sum_{d'} \eta_{nd'k'}} \tag{B.25}$$

Derivation for β

Taking the derivative of equation (B.5) with respect to $\beta_{kk'}$ subject to the constraint $\sum_{kk'} \beta_{kk'} = 1$:

$$\frac{\sum_{d,n} \phi_{ndk}\eta_{ndk'}}{\beta_{kk'}} + \lambda = 0 \tag{B.26}$$

$$\Rightarrow \beta_{kk'} \propto \sum_{d,n} \phi_{ndk}\eta_{ndk'} \tag{B.27}$$

Derivation for η

Taking the derivative of equation (B.5) with respect to $\eta_{ndk'}$subject to the constraint $\sum_{k'} \eta_{ndk'} = 1$:

$$\begin{aligned} & \textstyle\sum_f \log(P(R_{ndf}|Y_n = k, \mu'_{fk'}, \sigma'_{fk'})) \\ & + \textstyle\sum_k \phi_{ndk} \log \beta_{kk'} \\ & - [\log \eta_{ndk'} + 1] \\ & + \lambda \\ & = 0 \end{aligned} \tag{B.28}$$

$$\begin{aligned} \Rightarrow \eta_{ndk'} \propto \quad & \exp(\textstyle\sum_f \log(P(R_{ndf}|Y_n = k, \mu'_{fk'}, \sigma'_{fk'}))) \\ & \exp(\textstyle\sum_k \phi_{ndk} \log \beta_{kk'}) \end{aligned} \tag{B.29}$$