Data Projections for the Analysis and Visualisation of Bioinformatics Data

Karen Lees
Linacre College

Robotics Research Group
Department of Engineering Science
University of Oxford

Hilary Term 2008

This thesis is submitted to the Department of Engineering Science, University of Oxford, for the degree of Doctor of Philosophy. This thesis is entirely my own work, and, except where otherwise indicated, describes my own research.
Data Projections for the Analysis and Visualisation of Bioinformatics Data

Abstract

The development of high-throughput methods has lead to an explosion in the amount of data relating to molecular biology. The data sets are often large, noisy and multi-dimensional but the information hidden within them has the potential to increase the understanding of molecular pathways vital to biological processes, disease and development. It is impossible to decipher patterns from these large data sets manually, so the key to unlocking this information lies with automated methods that can extract or highlight these hidden patterns. It can be useful to use data projection methods, which project data onto a new set of variables, particularly if this new set of variables is much smaller or more meaningful for the interpretation of patterns. For instance, gene expression data can be projected into a data space that defines each gene’s association with certain expression patterns, or it can be projected into 2D to allow visualisation of the data on a 2D plot. This thesis reviews some data projection techniques along with how they have already been applied to gene expression data.

The thesis goes onto develop a new method for the analysis of gene expression data, involving two data projections, that decomposes the data into clusters based on relative changes in expression and from which some information about the underlying expression patterns in the data can be automatically mined. In doing so, this thesis shows how to use a mixture modelling approach with circular distributions. It also shows how gene expression data can be visualised using the GTM algorithm, how GTM can also be adapted for modelling circular data, and how results from variational Bayes mixture modelling can be used in conjunction with the GTM visualisations. Finally, the thesis suggests how matrix factorisation can be used to locate combinations of transcription factor binding sites present in the promoter regions of co-regulated genes.
## Contents

1 Introduction ............................................ 1
   1.1 Motivation ........................................ 1
   1.2 Contribution and Content ......................... 4

2 Gene Expression and Gene Regulation .................. 10
   2.1 Introduction ..................................... 10
   2.2 DNA, RNA, Genes and Proteins ................. 10
      2.2.1 Gene Expression ............................ 13
      2.2.2 Gene Regulation ............................ 16
   2.3 Microarray Technology ............................ 18
      2.3.1 Spotted Microarrays ......................... 18
      2.3.2 Affymetrix .................................. 19
      2.3.3 Microarray Data ............................ 21
   2.4 The Rice Blast Fungus Development ............. 23
      2.4.1 Rice Blast Microarray Data Set ............. 24
   2.5 Yeast Environmental Shock Response ......... 26
   2.6 Summary .......................................... 28

3 Review of Data Projection Techniques ................. 29
   3.1 Introduction ..................................... 29
   3.2 Visualising Multi-dimensional Data ............ 30
      3.2.1 Dimensionality Reduction ................. 31
   3.3 Clustering Techniques ............................ 42
      3.3.1 K-means Clustering ......................... 42
      3.3.2 Hierarchical Clustering ................... 43
      3.3.3 Self-organising Maps ....................... 45
      3.3.4 Correlation Graphs ......................... 45
      3.3.5 Mixture modelling .......................... 46
      3.3.6 Times Series Data Analysis ................. 55
   3.4 Data Transformations ................................ 55
   3.5 Summary .......................................... 55

4 Visualisation of Simple Data sets & Structural Analysis of Gene Expression Data .... 59
   4.1 Introduction ..................................... 59
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2 Simple Data sets</td>
<td>60</td>
</tr>
<tr>
<td>4.2.1 Linear Problem</td>
<td>60</td>
</tr>
<tr>
<td>4.2.2 S-curve Manifolds</td>
<td>62</td>
</tr>
<tr>
<td>4.2.3 Swiss Roll Manifolds</td>
<td>65</td>
</tr>
<tr>
<td>4.3 Visualisation of Clustered Data</td>
<td>69</td>
</tr>
<tr>
<td>4.3.1 1D Example</td>
<td>69</td>
</tr>
<tr>
<td>4.3.2 2D Example</td>
<td>71</td>
</tr>
<tr>
<td>4.4 Structural Analysis of Gene Expression Data</td>
<td>72</td>
</tr>
<tr>
<td>4.4.1 Pairwise Analysis of Conditions</td>
<td>74</td>
</tr>
<tr>
<td>4.4.2 Application of GTM to the microarray data</td>
<td>78</td>
</tr>
<tr>
<td>4.5 Summary</td>
<td>81</td>
</tr>
<tr>
<td>5 Angular Distribution Decomposition</td>
<td>84</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>84</td>
</tr>
<tr>
<td>5.2 Methodology: Data Projections</td>
<td>85</td>
</tr>
<tr>
<td>5.2.1 Projection to Angular Space</td>
<td>85</td>
</tr>
<tr>
<td>5.2.2 Mixture Modelling of Circular Data</td>
<td>86</td>
</tr>
<tr>
<td>5.2.3 Noise Model Implications for the Microarray Data</td>
<td>91</td>
</tr>
<tr>
<td>5.3 Methodology: Interpreting the Results</td>
<td>91</td>
</tr>
<tr>
<td>5.3.1 Locating Interesting Clusters</td>
<td>91</td>
</tr>
<tr>
<td>5.3.2 Analysis of Genes Within a Cluster</td>
<td>92</td>
</tr>
<tr>
<td>5.4 Application to Gene Expression Data sets</td>
<td>93</td>
</tr>
<tr>
<td>5.4.1 Yeast Stress Response Data Set</td>
<td>93</td>
</tr>
<tr>
<td>5.4.2 Rice Blast Data Set</td>
<td>99</td>
</tr>
<tr>
<td>5.5 Summary</td>
<td>105</td>
</tr>
<tr>
<td>6 Visualisation of Genetic Information on a 2D Map</td>
<td>107</td>
</tr>
<tr>
<td>6.1 Introduction</td>
<td>107</td>
</tr>
<tr>
<td>6.2 GTM for Angular Data</td>
<td>108</td>
</tr>
<tr>
<td>6.3 Visualisation of the Microarray Data sets</td>
<td>111</td>
</tr>
<tr>
<td>6.3.1 Visualisation of Patterns Using Cluster Information</td>
<td>115</td>
</tr>
<tr>
<td>6.3.2 Visualisation of Annotations</td>
<td>121</td>
</tr>
<tr>
<td>6.3.3 Visualisation of Selected Subsets</td>
<td>123</td>
</tr>
<tr>
<td>6.3.4 Visualisation of Substructure</td>
<td>126</td>
</tr>
<tr>
<td>6.4 Summary</td>
<td>126</td>
</tr>
<tr>
<td>7 Locating Combinations of Transcription Factor Binding Sites: A Matrix Factorisation Approach</td>
<td>128</td>
</tr>
<tr>
<td>7.1 Introduction</td>
<td>128</td>
</tr>
<tr>
<td>7.2 Methods</td>
<td>130</td>
</tr>
<tr>
<td>7.2.1 Motif Score Matrix</td>
<td>130</td>
</tr>
</tbody>
</table>
7.2.2 Decomposition of the score matrix using non-negative matrix factorisation ................................... 132
7.3 Results on Simulated Data ................................................................. 134
7.3.1 Data sets ............................................................................. 137
7.4 Results on Rice Blast Data ............................................................... 145
  7.4.1 Are there any motif combinations specifically associated with clusters
       A or B? ............................................................................ 146
  7.4.2 Do the groups of genes in cluster D discussed in section 6.3.2 share
       the same motifs? ................................................................. 149
  7.4.3 Are there any similarities or differences in the promoter regions of the
       secreted proteins? .............................................................. 155
7.5 Discussion ............................................................................... 156

8 Summary and Further Work ......................................................... 160
  8.1 Summary .................................................................................. 160
     8.1.1 Analysis of Gene Expression Data ........................................... 161
     8.1.2 Visualisation of Gene Expression Data .................................... 162
     8.1.3 Combinations of Transcription Factor Binding Site Motifs ........ 163
     8.1.4 Circular Data Analysis ........................................................ 164
  8.2 Further Work ........................................................................... 165
     8.2.1 Visualisation ................................................................. 165
     8.2.2 Motif Analysis ............................................................ 168
     8.2.3 Biological Research ......................................................... 171

I Appendix ................................................................................. 172
  A Circular EM and Circular VB Mixture Modelling ......................... 173
     A.1 Circular Expectation-Maximisation ......................................... 173
     A.2 Circular Variational Bayes ....................................................... 174
        A.2.1 Priors ........................................................................ 174
        A.2.2 Posteriors .................................................................. 175
        A.2.3 Update Equations ......................................................... 175
     A.3 Initialisation of Model Parameters ............................................ 177
        A.3.1 Circular EM .............................................................. 177
        A.3.2 Circular VB ........................................................... 178
     A.4 Comparison of Circular EM and Circular VB ....................... 178

  B Pre-processing of the Rice Blast Data set ..................................... 180
     B.1 Description of Microarrays .................................................... 180
     B.2 Pre-processing ................................................................ 180
Acknowledgements

First and foremost, I must thank Stephen Roberts and Sarah Gurr for their constant advice, support, encouragement and understanding over the past few years despite my bouts of grumpiness and negative thinking.

I thank the LSI DTC and EPSRC for organising and funding my DPhil programme, without the financial support and education they provided I could not have undertaken this research.

I would like to thank all the members of PARG with whom I have had many interesting discussions during my time in Oxford. In particular, thanks to Iead Rezek and Steven Reece for listening to me, offering advice, cups of coffee, taking me to the pub and reminding me that I really needed to write my thesis.

I must thank my parents for not being too shocked when I told them that I wanted to be a student for four more years and also for reminding me that I really needed to write my thesis.

I thank all the friends I made while studying in Oxford. In particular, I thank Jo Davies for her endless supply of tea, moral support and discussions on all things statistical. I thank Catherine Kelly for listening to my crazy ideas about biology over hot chocolate. Finally, a very special thanks goes to Duncan Lowne, a man who taught me that there was a huge exciting world outside the lab, that I should step outside it occasionally and go have some adventures.
Introduction

1.1 Motivation

Molecules such as DNA, RNA and proteins are contained within unicellular biological organisms and within every cell of multicellular organisms. The interplay between these molecules is vital to all cellular functions and processes. Without them, and a complex set of interaction pathways occurring between them, life as we know it would not exist. Development, cellular differentiation and genetic diseases occur directly as a result of sequenced or staged interactions of these molecules and the changing of the molecules during a cell’s lifetime.

Fundamental to the understanding of molecular biology, therefore, is the knowledge of which molecules are required for certain cellular functions and an understanding of the regulation of pathways that are essential to different cellular processes. Proteins are the molecules responsible for carrying out cellular functions and processes and the DNA in the cell encodes the information necessary to produce these proteins. A ‘protein coding gene’ is a sequence of nucleotide bases that encode instructions for the sequence of amino acids required to build a protein. Protein molecules are created when these instructions are ‘read’ in a process called gene expression and the more a gene is ‘read’, the more of the corresponding protein is produced in a cell. Only a small number of the genes in the genome will be expressed at a particular time, with genes turning on and off at different times during the cells lifetime in a process known as gene regulation. Gene expression and regulation involve molecular interactions between certain proteins and the DNA molecule(s) and are fundamental to all cellular activity.

“Traditional” experimental techniques in molecular biology, such as northern blot tran-
script analysis, have only provided insight into the regulation of a small proportion of the
genes within a given genome, even in relatively well studied organisms as they involve
many hours of laboratory work and only study single genes at once. Thus, data sets have
been small and differences in gene expression could be investigated by comparing the abund-
dances of a small number of molecules across a small number of experiments, without the
need for complicated statistical techniques. However, the regulation pathways of genes
and function of the proteins they encode has remained elusive because of the complex in-
terplay of molecules within cellular processes and the large number of potential protein
coding genes that may be involved and have yet to be studied.

Over the past decade, there have been major advances in experimental techniques, and
high-throughput methods have emerged. For example, microarray technology provides in-
f ormation about the expression of thousands of genes simultaneously, in some cases, all
the predicted genes in the genome. Advances have also been made in genome sequenc-
ing, the genome sequence of many different organisms has been analysed, from which the
protein coding regions have been predicted. As such, these technologies have drawn much
attention in our quest to understand biological processes at the molecular level and the
changes to the gene regulation and sequences that, for example, can cause disease or affect
development.

Unfortunately, simply having vast amounts of data does not provide an insight into biol-
ogy. Microarray data sets contain information about thousands, or tens of thousands of
genes across multiple experiments and are large, noisy and multi-dimensional. Genome
sequences are large, for instance the human genome contains over 30,000 genes in a se-
quence over 3 billion base pairs long. The interpretation of data on this scale is impossible
to do manually, yet it would be of immediate value to use the information from these data
sets to direct future research; allow patterns to be seen on a genome wide scale and im-
plicate genes with vital roles in disease and development. The ability to make predictions
about events at a sub-cellular level could have important implications to the future of drug
design and medical research. The valuable information in these data sets is, however, often
hidden and without automated ways to highlight the important information, the effort and expense invested in producing these data sets is wasted.

The main purpose of this thesis is therefore to investigate automated ways of finding patterns and highlighting features in the data, by modelling the data with a smaller, more manageable set of variables or parameters, from which biological interpretation is much easier. In this thesis a data set is viewed as a matrix of real numbers which may contain hidden structure and patterns that need to be discovered. There are many ways to manipulate and model data in order to reduce the information to a form that is more meaningful and more easily interpretable by humans. Dimensionality reduction techniques can be used to map the data to lower dimensional space; models of the distribution of data in multi-dimensional space can be used to locate groups of points; finding an alternative basis for the matrix can highlight underlying components as can transforming the data with some function. Reducing or changing the information in the data set to a different set of variables requires some thought as to which information in the data set should be treated as noise or redundancy and what is the potentially important information.

There are many types of microarray data sets. Some provide multiple repeats of experiments in two or more different conditions (e.g. cancerous, non-cancerous cells) and the aim is to rank the genes based on their differential expression between the two conditions. This thesis is mainly concerned with data sets that examine gene expression across multiple conditions, in which multiple comparisons of timepoints and/or treatments are required in the analysis and in which multiple genes are likely to show the same pattern of expression change. For instance, during development, there will be groups of genes that “turn on” and “off” at the same timepoints (co-regulated) in order to carry out specific biological processes and it would be interesting to discover the different patterns of expression change that occur.

In the absence of knowledge regarding which genes are co-regulated and what the true set of expression patterns are, verifying the results or comparing the quality of results from different methods is difficult. Microarray data is inherently noisy and as such is perhaps
1.2 Contribution and Content

not reliable enough to use to make definite conclusions about the regulatory patterns of specific genes but general patterns of expression may be deciphered, even in the presence of noise. The results of analysis can also guide researchers towards potential target genes for further investigation, which together with other information, can be used to select genes for researchers to concentrate their resources on. The analysis techniques presented in this thesis are therefore intended to provide information regarding expression patterns present in a data set, suggestions for target genes and hypothesis for gene regulation, but any conclusions about the functioning of particular genes should be verified with more rigorous experimental investigation.

1.2 Contribution and Content

The main theme of thesis is the application of data projections for the analysis of bioinformatics data. While the emphasis is more mathematical than biological, having a simple overview of what is currently understood about gene expression and gene regulation will aid the understanding of the data analysis done in this thesis. Thus, chapter 2 provides a brief overview of the biology connected with the data sets analysed. It summarises the mechanisms of gene expression and gene regulation; the relationship between DNA, RNA and protein molecules; what patterns of expression change mean and why biologists are interested in uncovering the patterns of expression associated with the cells of organisms across different environmental conditions. Microarray technology, which measures the gene expression for thousands of genes in a biological sample, is introduced and an explanation of how microarray data sets are created is given. One of the microarray data sets that is analysed in this thesis is that of the rice blast fungus, *Magnaporthe grisea*, a fungus which destroys a vast amount of the world’s rice crops each year. Chapter 2 introduces rice blast and why there is much interest in understanding the molecular biology of this cereal killer, particularly during infection related development.

In chapter 3, various mathematical techniques are reviewed for transforming, modelling and changing the dimensionality of data. All have a common theme of projecting data into
an alternative space, defined by a new set of variables. As previously explained, a main
aim of this thesis is to find patterns and highlight structure in the data, that can be de-
scribed by a smaller set of variables such that interpretation of patterns within the data
is easier. Dimensionality reduction techniques are reviewed, in particular with reference
to their potential for visualisation of patterns within the data. Clustering techniques are
discussed, with reference to how they locate patterns and how clustering has been previ-
ously beapplied to gene expression data. Transformations of gene expression data are also
used by the bioinformatics field; a review of the effect transformations have on the data
and how it affects the interpretation of data points and clusters is given. A critical review
is also given of the techniques already used for visualising and analysing gene expression
data, the assumptions they make, and how well they improve the understanding of gene
expression patterns hidden in the data. While some techniques may give reasonable mod-
els of the data under certain parameterisations, choosing appropriate parameterisations
is not always straight-forward and results can be quite complicated and difficult to inter-
pret. For example, when clustering data, deciding how many underlying patterns should
be searched for or how to interpret the key features of the expression patterns.

**Chapter 4** investigates visualisation of data using dimensionality reduction techniques
to reduce data sets to a set of two variables, such that the distribution of data can be
visualised on a 2D plot. To reduce the dimensionality from multi-dimensional to 2D re-
quires assumptions to be made about the data, for example in the form of a basis that
describes the variability in the data or assuming that the data points are distributed along
the surface of a 2D manifold. Principal component analysis (PCA), Locally Linear Embed-
ding (LLE) and Generative Topographic Mapping (GTM) are applied to simple data sets to
demonstrate the advantages and disadvantages of projecting data in to lower dimensional
space and what sort of structures the different methods highlight. The choice of param-
eters affects the results obtained from both LLE and GTM, but GTM is found to be less
sensitive to the particular parameter choice and noise in the data than LLE. It can also
model non-linear data, unlike PCA. A discussion is given on using GTM to model data that
is in the form of discrete clusters, rather than data spread on a continuous manifold. GTM can cope much better with clustered data than PCA or LLE, but a good choice for the number of basis functions or use of magnification factors are required to aid the visualisation of inter-cluster separation.

It goes on to investigate the structures in the microarray data sets to establish which models may be most appropriate for the visualisation of microarray data. By taking pairwise comparisons of the data for various conditions, a gene can be viewed as a datapoint positioned in 2D space, its angular dispersion from the centre describing a relative change in expression between the conditions and the radial dispersion from the centre providing information about the magnitudes of expression changes. If many genes show similar relative changes across a series of conditions, then these genes will be distributed in a ‘jet’ of data (angular structure) in multi-dimensional space. Examples of data plots showing pairwise comparisons of conditions are given in which these angular structures can be seen. Examination of GTM manifolds and magnification factors for data in a larger number of comparisons also suggests that these angular structures exist and a large percentage of genes are contained within a central hypersphere of data showing no regulatory change in any of the conditions. Applying GTM directly to these data sets therefore may not highlight these interesting structures. Instead, it may be advantageous to firstly remove the large number of genes that show little changes in expression and then model the genes in terms of their association with the angular patterns present in the data.

**Chapter 5** defines a method to model the distribution of angular data. Firstly, a transformation is defined, in which a series of pairwise comparisons of conditions are chosen, and then the data is transformed into angular data, by taking the angular dispersion for each pair of conditions. The effect of this transformation is compared with other transformations of the data and it is explained how the axes of the angular data space are more informative. The distribution of the angular data is modelled using a circular variational Bayes mixture modelling approach. The data is hence decomposed into clusters, based on the highest posterior probability of a gene with the mixture components. The clusters can
then be automatically interpreted with reference to relative changes in expression, and each gene visualised in terms of the association with the mixture component (and thus expression pattern) and the radial distance (which defines the amount of expression change). The circular variational Bayes approach (angular distribution decomposition) is applied to two microarray data sets, that of yeast adapting to environmental changes, which has been previously analysed by hierarchical clustering techniques; and a data set of infection related development in rice blast. This analysis of the yeast data set produced the two large clusters of genes associated with induction and repression, shown to be present in previous analysis, but was also able to locate smaller clusters, showing subtly different expression patterns. The analysis of the rice blast data identified clusters associated with pathogenic development. These clusters included some of the small number of genes already thought to be virulence factors as well as many more that had not been previously identified. Further analysis of genes in these clusters has been done to short list those most interesting for future research and knockout studies of one of the genes in these clusters showed impaired development of the rice blast fungus, such that it was no longer able to infect plant leaves.

Chapter 6 ties together work from previous chapters. Using the ideas of dimensionality reduction to project higher dimensional data into two dimensions and probabilistic clustering methods to find groupings of data points in the higher dimensional space, it shows how to use the results of GTM in conjunction with a variational Bayes mixture model to provide a more informative visualisation of the clusters projected into two dimensional space. For instance, the information from a variational Bayes or circular variational Bayes model can be used to identify the expression patterns associated with different regions of the GTM plot and other information such as functional annotations to be highlighted, to provide an overview of the distribution of genes associated with different functions across the data space and clusters. It also shows how to extend the GTM algorithm for use with circular data, and hence how the angular transformed data can be visualised using GTM. Chapters 4 to 6 demonstrate the use of data projections for the analysis and visualisation
of gene expression data, allowing clusters of genes to be located that share regulatory patterns (by some definition of similarity). Of more interest to biological researchers is why the genes seem to be co-expressed, for instance it is of interest to know if the genes are controlled by the same regulatory proteins and if similar sequence motifs are found in the genome close to these genes, which could act as binding sites for regulatory proteins. Thus, chapter 7 follows on from the previous chapters by searching for combinations of binding sites that are found in the genome upstream or downstream of the gene encoding regions of co-expressed genes. Previous techniques located individual sites in a single sequence but have a high false positive rate. As combinations of transcription factors are thought to work together to control transcription, some methods look for combinations of sites that occur within a given sequence window in a single sequence to reduce the false positive rate. Chapter 7 uses a data projection technique to search for combinations that occur multiple times within a set of co-regulated genes rather than searching through different possible combinations in a single strand in different windows. A set of genes may appear co-regulated across a small number of microarray experiments, but in fact be controlled by very different regulatory proteins, thus there may be multiple combinations of transcription factors associated with a cluster of genes identified from the gene expression analysis. It may also be possible that a cluster of genes identified from gene expression data analysis do not share any patterns of transcription factor binding sites and all have different regulatory pathways with respect to the set of binding sites that are being searched for. The data projection discussed in chapter 7 allows these hypotheses to be investigated, by dividing the information in a way that allows these sorts of patterns, or perhaps just as importantly, the absence of patterns, to be identified.

Chapter 8 summarises the main contributions of this thesis to the analysis of bioinformatics data and more generally to automated pattern finding in noisy multi-dimensional data sets. The developments to the variational Bayes mixture model and GTM can be used with any angular data set, the methodology is not restricted for use with gene expression data. The visualisation, clustering and motif analysis can be used independently or together as
a modular system of analysis in which clusters are found and then patterns and substructure are highlighted within each cluster. Future work and further improvements are also discussed.
Gene Expression and Gene Regulation

2.1 Introduction

Chapter 1 introduced some concepts in molecular biology, how functions depend on interactions between different cellular molecules and how data from high-throughput techniques may be useful in understanding more about these interactions. This chapter introduces these concepts in more detail, explains what microarray data is and how it measures gene expression. To demonstrate the use of finding expression patterns with microarray data two microarray data sets, which will be analysed throughout this thesis, are introduced.

2.2 DNA, RNA, Genes and Proteins

Deoxyribonucleic acid (DNA) is contained within the cells of living organisms. This nucleic acid stores the instructions necessary for a cell to function during its lifetime. A DNA molecule is a long polymer composed of a sugar (2-deoxyribose) and phosphate backbone with one of four bases attached to each sugar (figure 2.1; Berg et al. (2002); Alberts et al. (2002)). The four bases are adenine (A), cytosine (C), guanine (G) and thymine (T). Each unit in the DNA molecule is called a nucleotide. The sequence of these bases along a DNA strand encodes the information for the functioning of a cell. Prokaryotes such as bacteria store the DNA molecules within the cell cytoplasm but in eukaryotic organisms such as animals, plants, and fungi the DNA is found within the nucleus (Berg et al., 2002; Alberts et al., 2002). The DNA is organised into structures known as chromosomes (figure 2.2), the number of the chromosomes found in a cell varies depending on the type of organism the cell belongs to.

A DNA strand is 22 to 26 Ångstroms wide (2.2 to 2.6 nanometres) with each nucleotide unit
2.2 DNA, RNA, Genes and Proteins

Figure 2.1: Nucleic acids - DNA and RNA are long polymers composed of a sugar-phosphate backbone with nitrogenous bases attached to each sugar molecule. DNA usually has two strands bonded together forming a double helix structure, A bonds with T and C with G on the opposite strand. RNA usually exists as a single strand and has the base Uracil (U) rather than Thymine (T). Picture taken from the NHGRI Talking Glossary of Genetic Terms, available at http://www.genome.gov/10002096.

Figure 2.2: Chromosome - DNA is arranged in structures known as chromosomes. Each chromosome contains a single continuous piece if DNA and has other proteins associated with it to package the DNA molecule. For instance for instance it is supercoiled around histone octomers. Picture taken from the NHGRI Talking Glossary of Genetic Terms, available at http://www.genome.gov/10002096.
2.2 DNA, RNA, Genes and Proteins

Figure 2.3: Protein - a protein molecule is a long chain of amino acids. Picture taken from the NHGRI Talking Glossary of Genetic Terms, available at http://www.genome.gov/10002096.

being 3.3 Ångstroms (0.33 nanometres) long (Mandelkern et al., 1981). The nucleotides in DNA strands are therefore very small, but entire DNA strands can be enormous molecules, containing millions of these nucleotides. For instance, the largest human chromosome, chromosome number 1, is 220 million base pairs long (Gregory et al., 2006). DNA does not normally exist as a single molecule but as two molecules bonded together as a double helix (Watson and Crick, 1953; Alberts et al., 2002). The molecules are joined with hydrogen bonds that form between C and G molecules and between A and T molecules (known as Watson-Crick base pairing).

The DNA sequence for a genome is thus a sequence of A’s, C’s, G’s and T’s found across the set of chromosomes in the cell. Contained within this sequence are sections that encode the nature of proteins (protein coding genes). Protein molecules are chains of amino acids (figure 2.3) and vital to all cellular processes. There are 20 different amino acids, and the sequence of these in the protein give rise to the protein’s structure and function (Berg et al., 2002; Alberts et al., 2002).

Ribonucleic acid (RNA) is another important nucleic acid (figure 2.1). Different types of RNA exist, one of which is messenger RNA (mRNA), which helps the instructions encoded in the protein coding genes of the DNA to be converted into the corresponding proteins (Berg et al., 2002; Alberts et al., 2002). RNA is similar to DNA, but normally only exists as single strands and contains the base uracil (U) rather than thymine (T). The mechanism by which the instructions in a protein coding gene are ‘read’ and the corresponding protein
Figure 2.4: Gene expression - genes are expressed when their DNA sequence is transcribed into mRNA and then these mRNA molecules are translated into proteins. The transcription process produces a copy of the information in the gene, but in the form of an mRNA molecule that can move outside of the cell nucleus and be used to make a protein molecule. The translation process reads the sequence of bases in the mRNA molecules and converts them into a corresponding sequence of amino acids using the genetic code. Picture taken from the NHGRI Talking Glossary of Genetic Terms, available at http://www.genome.gov/10002096.

produced is referred to as gene expression and is explained further in the following section.

### 2.2.1 Gene Expression

Although a genome may contain many thousands of protein encoding genes, not all of these genes are expressed ('read') at the same time. Only those required for the functioning of the cell will be expressed. Therefore, even though different cells from the same organism may contain the same genetic information, it is the expression of particular genes that dictate the proteins present in a cell, the behaviour of a cell and the type of a cell. For instance, a blood cell is different to a skin cell because of the set of genes that are being expressed and thus the set of proteins it contains. The two major steps in gene expression are gene transcription and translation (figure 2.4).
2.2 DNA, RNA, Genes and Proteins

Figure 2.5: Codons - each triplet of 3 bases on the mRNA strand is called a codon and translated into a amino acid using the genetic code (table 2.1). Picture taken from the NHGRI Talking Glossary of Genetic Terms, available at http://www.genome.gov/10002096.

**Transcription (DNA → mRNA)**

During transcription the DNA molecule is unravelled in the region that encodes the protein and an mRNA molecule is then created containing the corresponding sequence of nucleotides (Berg et al., 2002; Alberts et al., 2002). The mRNA is basically a copy of the information contained within the DNA, but unlike the DNA, the mRNA molecule can move around the cell and into the apparatus that will allow it to be translated into a protein sequence. The more a gene is ‘read’, the higher the abundance of the corresponding mRNA molecules.

**Translation (mRNA → Protein)**

Translation is the process by which the nucleotide sequence of an mRNA molecule is ‘read’ and a corresponding protein is built (Berg et al., 2002; Alberts et al., 2002). Every triplet of three bases of the mRNA (known as codons, figure 2.5) is translated into an amino acid, or is used as a start or stop signal. The rules specifying which codons correspond to which amino acids is called the genetic code (table 2.1). The higher the mRNA abundance the more protein molecules will be produced.
<table>
<thead>
<tr>
<th>First Position</th>
<th>Second Position</th>
<th>Third Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>C</td>
<td>A</td>
</tr>
<tr>
<td>Phe</td>
<td>Ser</td>
<td>Tyr</td>
</tr>
<tr>
<td>Phe</td>
<td>Ser</td>
<td>Tyr</td>
</tr>
<tr>
<td>Leu</td>
<td>Ser</td>
<td>Stop</td>
</tr>
<tr>
<td>Leu</td>
<td>Ser</td>
<td>Stop</td>
</tr>
<tr>
<td>C</td>
<td>Leu</td>
<td>Pro</td>
</tr>
<tr>
<td>Leu</td>
<td>Pro</td>
<td>His</td>
</tr>
<tr>
<td>Leu</td>
<td>Pro</td>
<td>Gln</td>
</tr>
<tr>
<td>Leu</td>
<td>Pro</td>
<td>Gln</td>
</tr>
<tr>
<td>A</td>
<td>Ile</td>
<td>Thr</td>
</tr>
<tr>
<td>Ile</td>
<td>Thr</td>
<td>Asn</td>
</tr>
<tr>
<td>Ile</td>
<td>Thr</td>
<td>Lys</td>
</tr>
<tr>
<td>Met</td>
<td>Thr</td>
<td>Lys</td>
</tr>
<tr>
<td>G</td>
<td>Val</td>
<td>Ala</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Asp</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Glu</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Glu</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Glu</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Asp</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Glu</td>
</tr>
<tr>
<td>Val</td>
<td>Ala</td>
<td>Glu</td>
</tr>
</tbody>
</table>

Table 2.1: The standard genetic code - rules specifying which nucleotide triplets (codons) encode for which amino acid during the transcription process. Some amino acids can be coded for by multiple codons. Some codons are used to signal the termination of a coding region (UAA, UAG, UGA). AUG is part of the initiation signal as well as coding for methionine (Met) residues within a protein chain.
2.2.2 Gene Regulation

The production of proteins in a cell is controlled at various stages of the transcription and translation processes, however, much of the control is achieved in the transcription initiation stage.

Transcription Initiation

Protein coding genes are transcribed from DNA to mRNA by the action of a protein called RNA polymerase II, this protein unwinds the DNA and moves along one of the DNA strands recruiting nucleotides to the end of the growing mRNA molecule. RNA polymerase II itself does not distinguish between different regions of the DNA and is capable of binding to all sections of the genome. However, during transcription it forms part of a complex of proteins, known as the basal transcription apparatus (BTA; figure 2.6). It is the other proteins in this complex that are responsible for the selective recruitment of the RNA polymerase II to the transcription start sites (TSS) of protein coding genes, and all are therefore essential for transcription process (Berg et al., 2002; Alberts et al., 2002; Latchman, 2003). One of the proteins in the complex, TFIID, binds upstream of the transcription start site and is associated with binding to the TATA box motif (Berg et al., 2002; Alberts et al., 2002; Latchman, 2003), which is an AT-rich region found in a large number of promoter regions of eukaryotic genomes ≈30 base pairs (bp) upstream of the transcription start site. The other proteins in the BTA then bind in a particular order to form the complex of proteins, and eventually recruit the RNA polymerase II.

The binding of the BTA is essential to the transcription of genes, but by itself is only responsible for low levels of transcription and is not selective as to which genes it will transcribe, it just locates gene start sites, indicated by sequence motifs such as the TATA box. The selective regulation of genes occurs due to the interaction of other proteins, called transcription factors. These transcription factors bind to specific sequences, which are typically 5-10bp long in eukaryotes (Reichmann, 2002), in the promoter region of the gene (the region adjacent to the gene). They interact with each other and with the proteins in the
BTA. These transcription factors therefore have a direct effect on the gene transcription by helping to recruit the complex of proteins in the BTA and affecting the stability with which it binds to the TSS. The longer the time spent and more frequently the basal transcription apparatus is bound near the TSS of the gene, the larger the amount of transcription can occur. In summary, transcription factors affect the probability with which transcription of certain genes will be initiated. Transcription factors can interact to either increase the probability of transcription (activators) or decrease the probability of transcription (repressors). Multiple transcription factors may work in combination to achieve different levels of transcription initiation.

Figure 2.6: The basal transcription apparatus (BTA) - a complex of proteins that bind to the DNA near the transcription start site of a gene and helps to recruit the RNA polymerase II molecule which transcribes the gene into mRNA. The protein TFII D binds first, often to a TATA box sequence roughly 30bp upstream of the transcription start site. The proteins TFII A, TFII B, TFII F, RNA polymerase II and TFII E then bind to form a complex. Picture reproduced from Berg et al. (2002).
The Chromatin and Effect of Enhancers  In order for eukaryotic DNA molecules to fit compactly inside the cell, their length is compressed by being tightly bound to small proteins called histones. Eight histone molecules form a histone octomer, and the DNA is supercoiled around these octomers, causing it to become compacted. Studies have suggested that the chromatin structure is extremely important to gene regulation (Berg et al., 2002; Alberts et al., 2002; Latchman, 2003). The DNA molecule along with all the associated proteins, such as histones, is referred to as the chromatin. If the DNA is supercoiled around the histones then the basal transcription apparatus and transcription factors cannot bind to the promoter region of the gene and thus transcribed cannot be initiated. Therefore, unwinding of genes from the histones is vital for the replication and transcription of the DNA. Evidence suggests that only a small proportion of the genome is accessible to transcription factors at any time (Berg et al., 2002). There is also evidence that the chromatin structure can be altered by the binding of specific proteins to sequences known as enhancers (Berg et al., 2002; Alberts et al., 2002; Latchman, 2003). The binding of proteins to enhancers exposes different sections of the DNA so that they are no longer supercoiled around the histones and allows the BTA and transcription factors access to the genes in these regions.

2.3 Microarray Technology

Microarrays are a recently developed experimental method with which the gene expression of thousands of genes can be measured simultaneously and very quickly by observing the mRNA abundances associated with each of the genes.

2.3.1 Spotted Microarrays

A spotted microarray consists of many thousands of short, single stranded sequences. Each are complementary to the cDNA strand representing a single gene. These sequences are known as the probes and in cDNA spotted arrays will typically be 500-2500 base pairs long (Schena, 2003).

In a microarray experiment, the mRNA from a cell sample is reversed transcribed into
cDNA molecules that incorporate fluorescently tagged bases (see figure 2.7). The fluorescently labelled cDNA mixture is then exposed to the microarray so that the cDNA can bind (hybridise) to the strands attached to the microarray chip using Watson-Crick base pairing. The strands on the microarray are in vast excess to those in the fluorescently labelled mixture so that the amount of bound cDNA reflects the proportion in the mixture. Two samples can be hybridised to the same spotted array for comparison by allowing the two different samples to incorporate different fluorescent dyes. For instance, the control sample (e.g. healthy) can be labelled with a green dye (cy3) and the other sample (e.g. diseased) labelled with a red dye (cy5). Excess mixture that has not bound is then washed off, leaving the hybridised strands. Using a laser, the fluorescent molecules in the cDNA can be observed and the intensity for each gene can be measured. A greater intensity of the green dye on a spot implies that the control sample contained more mRNA for that gene and therefore the gene was being expressed more in the control sample. Similarly, a greater intensity for the red dye implies that the gene was being expressed less in the control sample. An example of an image from a spotted array experiment is given in figure 2.8.

2.3.2 Affymetrix

The Affymetrix technology is summarised here but explained in more detail by Knudsen (2002). Oligonucleotides are single stranded 15 to 70 nucleotide molecules made by chemical synthesis. The chip contains several hundred thousand squares of oligonucleotides, each square containing several million copies of the same oligonucleotide. Unlike the spotted arrays where each gene is only represented by one spot, up to 40 oligos can be used in the detection of a single gene. A section of a gene that is least similar to other genes is chosen, from this region 11-20 oligos are created as perfect matches. Then 11-20 mismatch probes are created that are the same as the perfect matches except that the central position is changed to the complementary nucleotide. The perfect matches for a gene contain different sequences so hybridisation is not uniform for all the oligos for a gene. Affymetrix has only one fluorochrome so two chips are required to compare a control sample with another
Figure 2.7: Hybridisation to microarray slide - genes are expressed when their DNA sequence is transcribed into mRNA and then translated into proteins. Spotted microarrays measure gene expression by measuring the abundance of the mRNA molecules corresponding to the different genes. This is done by reverse transcribing the mRNA molecules into cDNA with fluorescently tagged bases and then hybridising them to the probes of microarray slide.

Figure 2.8: An image from a scanned microarray downloaded from http://www.cs.unm.edu/~patrik/networks/data.html. Red spots imply that the gene was less active in the control sample, green spots are genes more active in the control sample and yellow implies that the gene had similar activity in both samples.
2.3 Microarray Technology

Figure 2.9: Example of spotted microarray experiment - in this example a reference sample is labeled with a green dye and samples from other conditions are labeled with a red dye. The reference sample and one of the other conditions are then hybridised to each of the arrays. The mRNA abundances in the different conditions can then be compared with the reference sample.

2.3.3 Microarray Data

The gene expression in different cell types or similar cells under different conditions can be compared using microarrays. Each experiment can be recorded on a different microarray, the intensities calculated and analysed for each gene across the different arrays (figure 2.9).

**Extraction of Fluorescence Intensity Extraction and Pre-processing**

A laser scanner is used to detect the fluorescent emissions from the dyes at each spot. Software is then used to compute the intensities of the fluorochrome(s) at each spot. The intensity values are then processed by software for background subtraction and signal to noise ratio.

Normalisation techniques should also be applied to the data to correct for dye bias and spatial bias on spotted arrays. Microarray data is often converted to log ratio expression values of the two dyes on each slide during the preprocessing step, so that down-regulation of genes are on the same scale as up-regulation. Using ratio values rather than log ratio values, means that up-regulation is indicated by values from from 1 to infinity, whereas down-regulation is squashed between 0 and 1. Applying the log transform allows the both
down-regulation and up-regulation to be on the same scale, with negative values implying
down-regulation, positive values implying up-regulation, and zero implying no difference
in regulation.

Once the expression data has been extracted from all of the microarrays then a gene ex-
pression matrix is produced. Rows represent genes, columns the different experimental
conditions/samples. The values in a gene expression matrix are usually the log ratio of
the gene expression levels. Rows of values are referred to as gene expression profiles and
columns are sample expression profiles. Thus, a typical gene expression matrix, $X$, for $N$
genomes across $D$ samples/conditions is given by:

$$X = \begin{bmatrix}
    x_{1,1} & x_{1,2} & \ldots & x_{1,D} \\
x_{2,1} & x_{2,2} & \ldots & x_{2,D} \\
    \vdots & \vdots & \ddots & \vdots \\
x_{N,1} & x_{N,2} & \ldots & x_{N,D}
\end{bmatrix}$$

where $x_{ij} = \log_2 \left( \frac{R_i^j}{G_i^j} \right)$, $R_i^j$ is the intensity from the red channel for the spot corresponding
to the $i$th gene in the $j$th microarray slide and $G_i^j$ is the intensity from the green channel
for the spot corresponding to the $i$th gene in the $j$th slide.

**Analysis**

There are different types of microarray data sets. In some data sets classifications (e.g.
healthy, diseased) for the samples are known and the aim is to find the genes that seem
to be differentially expressed between the classifications. Where there are experimental
replicates, statistical tests such as the t-test and ANOVA have been used to mine the data
set for differentially expressed genes. The null hypothesis in these tests is that there is
no difference in expression between conditions. The t-test looks at the mean and variance
of log-fold change data from two different conditions and calculates the probability that
they were sampled from different distributions. Many variations of the test statistic have
been used such as the global t-test statistic, the gene specific test statistic, the 'significance
analysis of microarrays' (SAM) and the regularised t-test statistic. However, the t-test can
only be used when there are two different conditions, analysis of variance (ANOVA) has
been used for data sets with more than two conditions. Like the t-test there are many different variations of the test statistic that can be used in ANOVA. More details on using these test statistics for microarray data can be found in Cui and Churchill (2003). Gene shaving (Hastie et al., 2000) is an alternative method for locating genes with differential expression that finds clusters of genes showing high variation in expression across samples but are highly correlated across the genes. Gene shaving uses principal component analysis (PCA) to find ‘eigengenes’ that explain the largest amount of variation across the samples and correspond to patterns of differential expression.

Other data sets do not have class labels but look at patterns of gene expression across different conditions or time points. Techniques to analyse this type of microarray data have concentrated on clustering genes according to their expression patterns. These are discussed more thoroughly in chapter 3.

### 2.4 The Rice Blast Fungus Development

The rice blast fungus, *Magnaporthe grisea*, is renowned for its devastating effects on rice crops. Moreover, the fungus infects 750 grass species. Figure 2.10 shows the effect of the disease on a rice plant leaf. It destroys enough rice each year to feed 60 million people (Dean et al., 2005) or 10 – 30% of the world’s rice harvest (Talbot, 2003). It is of particular interest to biological researchers because half the world’s population rely on rice as food source, as well as being a model organism which to study plant-microbe interactions (Dean et al., 2005).

Essential to the infection process of this fungus is the development of the specialised infection structures known as the appressorium. The appressorium is a swollen cell, which emerges at the distal tip of the germinating asexual spore and then becomes caged in a veneer of cell-wall associated melanin. The compatible solutes within the appressorium change into 3M glycerol and a huge turgor pressure of up to 8MPa builds up (Dean et al., 2005). The pressure in the appressorium forces a penetration peg to emerge, 1µm in radius, which punctures through the leaf surface, allowing invasive growth to occur and then the
eventual death of the rice plant (figure 2.11). This pattern of development requires a hard hydrophobic surface (such as a plant leaf) and can be mimicked on artificial hydrophobic surfaces (figures 2.12 and 2.13). However, it has been noted that these infection structures do not develop when the fungus germinates on a hydrophilic surface, the germ tubes develop but the fungus cannot form an appressorium. Thus, the surface type and interaction of the fungus with the surface plays an important role in signalling the onset of infection related development.

Signalling is achieved by surface proteins and transduced by others intracellularly (by the movement of proteins across the cell membrane), which trigger a reaction cascade that drives morphogenesis. Understanding the gene regulation involved in the development of these infection related structures is key to understanding the infection process. Proteins secreted by the fungus and expressed in rice plant leaves, respectively, are involved in the attack and defence mechanisms. Certain proteins have already been identified in fungi as being involved in infection processes, and similar proteins identified in the rice blast genome by sequence analysis (Talbot, 2003). For instance, cutinases, which are proteins secreted by the fungus and used to break down the rice leaf surface prior to penetration, are known to be involved in the infection process (Talbot, 2003; Dean et al., 2005). Conserved fungal specific extracellular membrane spanning domain proteins (CFEMs) are involved in receiving signals from the environment and polyketide synthases (PKS) are a group of toxins involved with killing plant hosts (Dean et al., 2005). These, however form only a small percentage of the thousands of proteins potentially encoded in the rice blast genome, about which little is known.

**2.4.1 Rice Blast Microarray Data Set**

This data set followed changes in transcript abundance levels of 13,666 expressed sequence tags (ESTs) from the fungus *Magnaporthe grisea* during infection-related development. The five conditions of study in this microarray data set represent (1) the ungerminated spore sample, (2) at 7 hours and (3) 12 hours post inoculation on a hydrophobic surface and (4,5) the same time points post inoculation on a hydrophilic surface as detailed in Dean
2.4 The Rice Blast Fungus Development

Figure 2.10: A plant leaf infected with rice blast disease.

Figure 2.11: Rice blast asexual lifecycle. A fungus spore attaches to the surface of a plant leaf and germinates, during which a germ tube grows. At 7 hours after attachment, an appressorium (swollen cell) forms, in which huge turgor pressure builds up. At 12 hours a penetration peg emerges from the appressorium and punctures through the leaf. Infection and invasive growth then occur, new spores are formed and the lifecycle repeats.

Figure 2.12: Rice blast appressorium - electron microscopy of a rice blast fungus, showing the germ tube and appressorium (10\(\mu\)m wide). The appressorium grows on hydrophobic but not hydrophilic surfaces.
2.5 Yeast Environmental Shock Response

It is important for cellular organisms to maintain specific internal conditions in order to grow and function optimally (Gasch et al., 2000). The yeast *Saccharomyces cerevisiae* is capable of rapidly adapting to drastic changes in environmental conditions by changing its expression program to produce proteins that will help it to adjust to these conditions. The microarray data set studied by Gasch et al. (2000) observed patterns of expression change after various environmental shocks had been applied. The experiments included time courses of heat shock, osmotic stress, hydrogen peroxide treatment, nutrient starvation, acidity and response to toxic chemicals.

The mechanisms of response are quite complex. For example, studies suggest responses to certain stresses, such as exposure to high temperatures induce the transcription of proteins that encode heat shock proteins (Craig, 1992). Analysis of the data set by hierarchical clustering (Gasch et al., 2000) highlighted a large number of genes (∼ 900 of which ∼ 300 induced, ∼ 600 repressed) that showed a drastic response to all the different types
Figure 2.14: Rice blast microarray slides. Four slides compared the spore sample with other conditions post inoculation. (1) 7 hours hydrophobic surface (appressorium development) (2) 12 hours hydrophobic surface (penetration peg emergence) (3) 7 hours hydrophilic surface (remains in germinated state with no appressorium development) (4) 12 hours hydrophilic surface (remains in germinated state with no appressorium development or penetration peg emergence)
of environmental shock, termed environmental stress response (ESR) genes. The groups of induced genes probably include the genes necessary for the cell to maintain internal conditions and repressed genes are non-essential genes “turned off” in order to divert resources while the cell copes with the suboptimal conditions. There are also thought to be genes involved in response to more specific shocks.

### 2.6 Summary

Molecular biology depends on a complex sequence of DNA-protein, RNA-protein and protein-protein interactions. DNA sequences within protein coding genes as well sequences in the DNA that act as binding sites for regulatory proteins are also vital to the production of proteins. Mutations in these DNA sequences can disrupt the production of proteins, causing impaired cellular function and disease. Information about the gene transcription occurring in cells under different conditions can therefore help us to infer proteins required for certain biological processes and how they interact. Examining regions of the DNA sequence close to the gene coding regions of co-regulated genes may help us identify transcription factor binding sites.

Microarray technology provides a measure of the gene expression occurring in thousands of genes in a biological sample. This gives rise to large data sets in which patterns of expression change occurring across many experimental conditions are hidden. A motivating example is discussed of a microarray data set of the rice blast fungus, from which it would be of great value to biological researchers to identify genes involved in the infection process.
3

Review of Data Projection Techniques

3.1 Introduction

Chapter 2 reviewed the biology of gene expression and the creation of a microarray data set, which for the purposes of analysis can be thought of as a $N \times D$ matrix of log expression changes. In microarray data sets which examine the expression across different conditions or time points, the aim of the analysis is often to discover the underlying patterns of expression change and locate genes that are of most interest for future research. The data sets are large, multi-dimensional, noisy and impossible to analyse by hand. Data projection techniques project data to different data spaces, where the axes of the new data space correspond to a different set of variables. Projection of data into a different space can be advantageous for the interpretation of patterns within the data if this new set of variables is more meaningful or if the variable set is much smaller in size. Existing techniques for data projection, in particular dimensionality reduction techniques and clustering algorithms are introduced in this chapter, along with a review of how they have already been applied to gene expression data sets. Throughout the chapter we show how different methods project a $N \times D$ data matrix, $X$, into a new matrix $Y$ of dimensions $N \times d$.

In this chapter the projection techniques are separated into two groups. Firstly, those that can be used for the visualisation of gene expression data are reviewed. Humans are very good at locating patterns in low dimensional data, particularly when visualised in a two-dimensional diagram of some kind. Visualisation, therefore, simply finds a way of projecting data into 2D, and the pattern extraction is left to the human. Secondly, clustering techniques are reviewed. These find groups of data points that are similar in some sense and the data points are therefore projected into a data space which defines the association...
of the data points with the different groups. The different groups in the data may describe different expression patterns. So these projections allow the pattern extraction to be done automatically, by calculating each data point’s association with each of the patterns and partitioning the data accordingly.

3.2 Visualising Multi-dimensional Data

Gene expression data is usually visualised as either expression profiles across a series of experiments (see figure 3.1) or as a heat map (see figure 3.2). Heat maps display the expression ratios as a matrix of coloured squares. Genes are normally along the rows and the samples along the columns. Green represents down-regulation, red up-regulation and black no regulation change. The brighter the intensity of the green (or red) the greater the down (or up) regulation. They are often used in conjunction with a dendrogram of a hierarchical clustering tree, which groups similar profiles and make it easier to visualise patterns that occur in multiple genes. The plotting of expression profiles is also usually done in conjunction with some clustering technique, such that each cluster is plotted separately, as it would be difficult to see the different patterns if all the genes are plotted together.

Plotting the profiles for each cluster can be useful to visualise the patterns that have been clustered. However, for large clusters it can be difficult to see similarities and differences between genes within the cluster and to put a biological meaning to the cluster, particularly if a shaped based distance measure has been used for clustering. It is also difficult to see how similar or different two clusters are. Heat maps can be very useful for showing patterns within the entire data set to aid visualisation of groups in the heatmap, but even with a dendrogram they can become very large and difficult to interpret, with the squares representing each gene becoming very small. It is also difficult to visualise extra information about the genes with these visualisations, for instance if it was of interest to visualise the distribution of genes with the same functional annotation across the different expression patterns.
If it is possible to project the distribution of expression patterns into 2D space, then the expression of the genome across the conditions of study could be visualised as a point in a 2D map. Groups of genes showing similar expression patterns would correspond to areas of high data density. Thus it would be possible to view the expression change of the entire data set on a single plot, and similarities between different patterns (clusters) to be visualised as well as the similarities between data points. Ontological or functional information could be easily highlighted on this ‘genome expression map’, to give an overview of how the expression changes for genes with similar function differ. This type of visualisation could therefore provide a novel insight into the patterns of expression change. The projection to 2D space and incorporation of biological information could be easily automated. This section therefore reviews some dimensionality reduction techniques and how they can be used to project multi-dimensional data into 2D space.

3.2.1 Dimensionality Reduction

Often data is represented by a large number of measurements (variables), these are measurements that can be physically made by humans about the properties of an object under investigation. However, it is likely that a combination of the variables explain the characteristics under investigation. For instance, if one was trying to infer what characteristics are required to be a good athlete then various measurements can be taken. These may include height, weight, age and average speed. It is likely that a combination of some of the variables are important, whereas some variables will not be useful in explaining the characteristics of a good athlete. The number of measurements, $D$, taken for each athlete is the dimensionality of the problem, as the data for each athlete can be represented as a point in $D$ dimensional space. It is hard to visualise and therefore make sense of the data in dimensions higher than 3 and so it is useful to reduce the dimensionality of the data such that it is described in a more compact way, that visualises the relevant features. Dimensionality reduction therefore involves mapping a data set of $D$ variables into a data set of $d$ variables where $d < D$. There are various techniques for projection the data into $d$-dimensional space, each making different assumptions about what is important in the
Figure 3.1: Visualisation of data by plotting expression profiles - example of a microarray data set containing log fold changes for genes across 25 experimental conditions. K-means clustering has been used to separate the data into 12 clusters and the data for each cluster has been plotted separately.
3.2 Visualising Multi-dimensional Data

Figure 3.2: Visualisation using a heatmap. In this example the rows have been ordered according to hierarchical clustering result. The dendrogram shows the hierarchical clustering tree. Green is used to show down-regulation, red indicates up-regulation, and black indicates no change in regulation. Heat maps could also be used to show the expression profiles for each cluster in a $K$-means result and are not restricted to hierarchical clustering.
data and what can be treated as redundancy or noise. For visualisation purposes $d$ should be no greater than three.

In gene expression data the expression ratios of $N$ genes are measured under $D$ different conditions. Therefore each gene can be thought of a point in $D$ dimensional space positioned according to its intensity ratios in each experiment.

Matrix Factorisation Approaches

Matrix factorisations can be used in order to change the dimensionality of the data, by decomposing the original data matrix such that $X = UV$, where $U$ is an $N \times R$ matrix and $V$ is an $R \times N$ matrix:

$$
\begin{bmatrix}
x_{1,1} & \ldots & x_{1,D} \\
\vdots & \ddots & \vdots \\
x_{N,1} & \ldots & x_{N,D}
\end{bmatrix}
= 
\begin{bmatrix}
u_{1,1} & \ldots & u_{1,R} \\
\vdots & \ddots & \vdots \\
u_{N,1} & \ldots & u_{N,R}
\end{bmatrix}
\begin{bmatrix}
v_{1,1} & \ldots & v_{1,D} \\
\vdots & \ddots & \vdots \\
v_{R,1} & \ldots & v_{R,D}
\end{bmatrix}
$$

The rows of $V$ can be thought of as specifying a new set of $R$ variables that are linear combinations of the original variables (dimensions) of the data. $U$ specifies the data points in terms of this new set of variables. Thus, if $R < D$ the dimensionality of the data can be reduced simply by using the matrix $U$, which specifies the data points in terms of the smaller set of variables. There are multiple solutions to $X = UV$, thus certain assumptions have to be made in order to solve for $U$ and $V$. Different matrix factorisation techniques work by using different assumptions.

**Principal component analysis (PCA)** (Pearson, 1901; Jolliffe, 2002) finds a new basis for the data that specifies the axes of maximum variation of the data and such that the axes of the new basis are orthogonal. It has been shown that a suitable solution for the new basis, $B$, is given by the eigenvectors of the covariance matrix of $X$, ordered such that the first column (first principal component) corresponds to the eigenvector with the largest eigenvalue. Thus, $U$ and $V$ can be calculated by $V = EB^T$ and $U = XE^T B$, where $E$ is a diagonal matrix, whose diagonal elements give the eigenvalues corresponding to the
3.2 Visualising Multi-dimensional Data

Figure 3.3: Simple example of PCA reducing a 2D data set to 1D. The principal components are found, the data is then rotated and scaled so the principal components form the new axis. The dimensionality can be reduced to 1D by only using the first principal component.

eigenvectors in $B$. $U$ is thus effectively a rotation of the data such that the axes show the maximum variability of the data. The dimensionality of the data can be reduced by using only the first $d$ principle components to describe the data. Thus, if $B_L$ is the first $d$ columns of $B$, then a projection of $X$ into a lower dimensional space $Y$, can be achieved by $Y = XB_L$. By also dropping the multiplication by the eigenvectors in $E$, then $Y$ gives a version of the rotated data in which all axis have equal variance. If $d = 2$, then $Y$ provides a projection of the data into 2D space and can be used for visualisation. A simple example of PCA can be seen in figure 3.3.

**Independent Component Analysis (ICA)** models the data $X$ as being produced from a set of sources, which are defined in the rows of $V$ as linear combinations of the original dimensions, and a mixing matrix $U$, which specifies how the sources have been mixed to produced the data in $X$. $U$ and $V$ are chosen such that the sources $v_i$ are independent, which is equivalent to minimising the mutual information between them (Comon, 1994). The independence assumption means that ICA also assumes non-Gaussianity in the data, unlike PCA, which assumes Gaussianity. If $R < D$ then $U$ is a description of the data with reduced dimensionality. However, ICA cannot be used to project the data in 2D space, unless $R = 2$ or some choice is made about which are the two most important sources to visualise.
Non-negative Matrix Factorisation (NNMF) (Lee and Seung, 1999, 2001) uses the constraints that $X, U$ and $V$ are non-negative, to find a solution that minimises $X - UV$, the error in the approximation of $X$ given by $UV$. Again, if $R < D$, then the dimensionality of the data can be reduced by using $U$ as a description of the data in terms of a smaller set of variables, but there is no principled way of ordering the importance of the variables to select two most important for visualisation.

Locally Linear Embedding (LLE)

LLE (Roweis and Saul, 2000) projects data on a lower dimensional manifold by preserving the local neighbourhood of datapoints in some higher dimensional space. An advantage of this method over other local clustering algorithms is that it maps all the data points into the same lower dimensional coordinate system. Unlike methods such as Isomap (Tenenbaum et al., 2000) it does not try to preserve the distances between widely separated datapoints, only $K$ nearest neighbours but LLE can still recover the global nonlinear structure using only these local fits. The idea behind this is that if sufficient data has been sampled from the underlying manifold each point will be on a local linear patch with its neighbours. The local regions are characterised by linear coefficients that reconstruct each data point from its $K$-nearest neighbours using a weight matrix, $W$. After the weight matrix has been estimated, an error function (see equation 3.1) calculates how well the weights construct the data and the weight matrix is iteratively re-estimated until the error is minimised.

$$\xi(W) = \sum_{i=1}^{N} |X_i - \sum_{j=1}^{K} W_{ij}X_j|^2$$  (3.1)

The element $W_{ij}$ of the weight matrix is the contribution of the jth data point to the reconstruction of the ith data point. As each data point is only reconstructed from its nearest neighbours $W_{ij} = 0$ when $j$ is not considered a neighbour of $i$. If the data lies on a smooth non-linear manifold of lower dimensionality such that $d < D$ then there is a linear mapping that maps the higher dimensional coordinates to global internal coordinates of the manifold. It is expected that the local geometry in the lower dimensional space is the same.
3.2 Visualising Multi-dimensional Data

Figure 3.4: Example of the LLE working on an s-curve manifold. (a) is the manifold. (b) shows the sample points from the 3D manifold. (c) is the 2D LLE mapping of the data points from (b) as that of the higher dimensional space, so the weights that reconstruct data points in the high dimensional space can also be used in the low dimensional space. The lower dimensional mapping is therefore found by minimising another cost function $\Theta$, given in equation 3.2.

$$\Theta(Y) = \sum_{i=1}^{N} |Y_i - \sum_{j=1}^{K} W_{ij}Y_j|^2$$  

where $Y_i$ is the vector of coordinates in the lower dimensional space. In this equation the weight matrix is fixed and the values $Y_i$ are changed until the cost function is minimised. The value of $Y$ that therefore minimises this function is taken as the set of global internal coordinates to the manifold that optimally preserve the relationship between the data points.

There is only one free parameter in this algorithm which is the value of $K$, specifying the $K$ nearest neighbours to reconstruct each data point from.

**Examples** Two example non-linear problems for LLE are those of the s-curve and swiss roll manifolds. PCA would not be able to map this non-linear manifold, however given a well sampled data set, LLE can correctly group together points in the same area of the manifold and essentially unravel it into a lower dimensional space. Figures 3.4 and 3.5 show good results from the LLE algorithm on these non-linear problems.

**Advantages** LLE allows non-linear relationships to be preserved in a single coordinate system and is not limited to data of low dimensionality. There is no need to compute the
3.2 Visualising Multi-dimensional Data

Figure 3.5: Example of the LLE working on a swiss roll manifold. (a) is the manifold. (b) shows the sample points from the 3D manifold. (c) is the 2D LLE mapping of the data points from (b) distances between all the data points, only local distances are computed, which means this algorithm is computationally less expensive than some other methods. It is possible to estimate the best value of \( d \) by analysing a reciprocal of the cost function which gets weights from the embedding space to apply to the data points \( X_i \). Also this method requires only one free parameter which is fewer than most other dimensionality reduction techniques.

**Disadvantages**  LLE does not give a mapping of points onto the manifold, it only maps points relative to each other. Therefore if a new point is added to the data set then the entire mapping will have to be recalculated. The method requires lots of memory with large data sets and takes a long time to calculate the lower dimensional co-ordinates. For LLE to work well it requires that the data set lies on a continuous well sampled manifold and a suitable parameter value to be used. However it is not easy to choose a value for the parameter \( K \) as choosing either a value too small and too large produces poor results or to be certain that a data set is well sampled.

**Generative Topographic Mapping (GTM)**

GTM (Bishop et al., 1998) models the distribution \( p(X) \) of data in \( D \)-dimensional space in terms of \( d \) latent variables. If the latent space is two dimensional \( (d = 2) \) then it is like having a rubber sheet which the GTM algorithm attempts to stretch, compress and crumple locally in order to fit the data, \( X \), in the higher dimensional space. GTM assumes therefore that the the data lies on a \( d \)-dimensional manifold that may have been crumpled into a very complex shape in \( D \)-dimensional space. The manifold will be compressed around ar-
3.2 Visualising Multi-dimensional Data

... eas of high data density and stretched around areas of low data density, but the algorithm works by defining this $d$-dimensional manifold and projecting it into $D$-dimensional data space. Note that in order to keep notation consistent within this review chapter, some of the notation has been changed from that originally used to describe GTM in Bishop et al. (1998).

The latent space is defined by a grid $G$ of $K$ points, where each point $g$ in the grid is mapped to a point in $D$-dimensional space through a mapping function $h(g; W)$. This mapping is adjusted until the maximum likelihood solution of the data given the model parameters is found. The mapping can be implemented as a radial basis function (RBF) network in which $h(g; W) = W \phi(g)$ with a set of $M$ radially symmetric basis functions $\phi$ (see equation 3.3).

$$
\phi_m(g_i) = \exp \left\{ -\frac{\| g_i - \mu_m \|^2}{2\sigma^2} \right\}
$$

(3.3)

In order to find an expression for the maximum likelihood of the data given the model the following theory is used. The sampled points, $g_i$, from latent space are mapped to points $h_i$ in the data space, these points form the centres of radially-symmetric Gaussian distributions with variance $\beta^{-1}$. Therefore the probability distribution of a data point $x_n$ from a sampled latent point $g_i$ is given by equation 3.4.

$$
p(x_n|g_i, W, \beta) = \left( \frac{\beta}{2\pi} \right)^{D/2} \exp \left\{ -\frac{\beta}{2} \| h(g_i; W) - x_n \|^2 \right\}
$$

(3.4)

The distribution of the data point $x_n$ over the entire data space can then be found by integrating over all $g$ (equation 3.5).

$$
p(x_n|W, \beta) = \int p(x_n|g, W, \beta)p(g)dg.
$$

(3.5)

By defining $p(g)$ to be a sum of delta functions centred on the nodes of the grid in latent space (equation 3.6) the integration in equation 3.5 can be performed analytically and the probability distribution reduces to equation 3.7.
\[ p(g) = \frac{1}{K} \sum_{i=1}^{K} \delta(g - g_i) \quad (3.6) \]

\[ p(x_n|W, \beta) = \frac{1}{K} \sum_{i=1}^{K} p(x_n|g_i, W, \beta) \quad (3.7) \]

The maximum log likelihood solution to this is then found by maximising the log likelihood over all data points in \( X \) (equation 3.8) by adjusting the parameters \( W \) and \( \beta \) to find the optimal values \( W^* \) and \( \beta^* \).

\[
L(W, \beta) = \ln \prod_{n=1}^{N} p(x_n|W, \beta) \\
= \sum_{n=1}^{N} \ln \left\{ \frac{1}{K} \left( \frac{\beta}{2\pi} \right)^{D/2} \exp \left\{ -\frac{\beta}{2} \| h(g_i; W) - x_n \|^{2} \right\} \right\} \quad (3.8)
\]

If the mapping function \( h(g; W) \) is smooth and continuous then the projected points in the data space should be in the same ordering as the corresponding points in the latent space, so points close together in latent space should be mapped as close together in the data space. Although the data points in latent space form a grid of points equidistant from each other, the centres of the Gaussian distributions in data space will not necessarily show the same grid like distribution. The centres of the Gaussians will densely populate areas of the manifold that have a high density of data points.

**Data Visualisation with GTM**  The GTM algorithm maps a grid in latent space to points in the data space, however for visualisation of data sets in lower dimensions the opposite mapping is required. Bayes’ theorem can be used to invert the transformation such that data points in data space can be projected into latent space by using the mean or mode of the distribution for \( p(g|x_n, W^*, \beta^*) \). Thus, \( Y_n = \frac{1}{K} \sum_{i=1}^{K} g_i p(g_i|x_n, W^*, \beta^*) \).

**Magnification Factors**  The magnification factors describe the stretching and compressing of the latent space when embedded in the data space. The manifold will be stretched
3.2 Visualising Multi-dimensional Data

in regions of low data density and compressed in regions of high density. Magnification factors can be calculated as the ratios of infinitesimal areas in the latent space ($A_g$) to the corresponding area in the data space ($A_h$). Examples of the magnification factors and the stretching of the manifold are shown in figure 3.6. The magnification factor is given by equation 3.9 (Svensén, 1998).

$$\frac{dA_h}{dA_g} = \sqrt{\left\| \frac{\partial h}{\partial g_1} \right\|^2 \left\| \frac{\partial h}{\partial g_2} \right\|^2 - \left( \frac{\partial h_1}{\partial g_1} \frac{\partial h_2}{\partial g_2} \right)^2}$$  (3.9)

The partial derivatives of the mapping $h(g, W)$ with respect to the latent variable, $g$, are obtained from equation 3.10.

$$\frac{\partial h}{\partial g^j} = \psi_j W$$  (3.10)

where $\psi_{jm} = -\phi_m(x)(x^j - \mu^j_m)\sigma^{-2}$.

Figure 3.6: Examples of GTM manifolds and the associated magnifications factors. Manifolds and data points for a (a)s-curve data set and (b) swiss roll data set. Magnification factors for the (c)s-curve and (d) swiss roll manifolds.

**Advantages** The method actually defines a mapping between data space and latent space, so new data points can be placed in the latent space without the need to recalculate the mapping. It is a lot faster on large data sets and requires a lot less memory than
3.3 Clustering Techniques

**Disadvantages** There are five parameters to specify in the model which are number of basis functions, number of latent variables, width of basis functions, weight regularisation factor and the number of training cycles. It is not clear what combinations of these parameters would work best with a given data set. In order to use GTM to help visualise high dimensional data there is an assumption that the data lies on an 2D or 3D manifold which may not be true for complex data sets.

### 3.3 Clustering Techniques

Each gene in a microarray data matrix can be thought of as a point in $D$-dimensional sample space. Genes with similar expression patterns will therefore be close in the sample space and genes that have dissimilar expression profiles will be further apart. Clustering of gene expression data therefore involves finding groups of genes that are close together according to some distance measure, in order to identify genes that have similar changes in expression across sets of experimental conditions/samples. Each cluster can then be thought of as corresponding to an underlying expression pattern. Several classic techniques have been used to cluster gene expression data, including hierarchical clustering (Eisen et al., 1998), $K$-means clustering (Tavazoie, 1999), self-organising maps (Tamayo et al., 1999), independent component analysis (Liebermeister, 2002) and mixture models (Yeung et al., 2001; Ghosh and Chinnaiyan, 2002; McLachlan et al., 2002; Medvedovic et al., 2004; Muro et al., 2003; Teschendorff et al., 2005; Vogl et al., 2005). These techniques are compared and how they project data into an alternative data space in order to identify underlying patterns reviewed.

#### 3.3.1 K-means Clustering

K-means clustering (MacQueen, 1967) is a method that clusters $N$ data points into $K$ disjoint subsets by minimising an objective function (figure 3.7(a)). The algorithm is implemented by initialising $K$ centroids in the data space and then assigning each data point
to the cluster that has the closest centroid according to the chosen distance measure. The positions of the centroids are then re-calculated from the new cluster assignments. The assigning of data points to clusters and the re-calculation of the centroids is repeated until the centroids no longer move. A simple objective function to minimise in the K-means algorithm is sum of distances between data points and the cluster centroid. It can be thought of as projecting the data into $K$ subsets, or alternatively into $K$-dimensional space, where each datapoint is positioned at the co-ordinate 1 in one of the dimensions and 0 in all other dimensions.

Although the K-means algorithm will always terminate it is not guaranteed to find the optimal clustering result but only a local minimum for the objective function. The algorithm can also be sensitive to the initial placement of the centroids. To overcome these problems the algorithm can be run several times with different initialisations.

### 3.3.2 Hierarchical Clustering

In hierarchical clustering ((Johnson, 1967); figures 3.7(b) and 3.8) a distance measure is chosen and the data points are successively grouped together into a hierarchical tree structure. Pairwise distances are calculated between objects, the closest two are then grouped together. The objects can be individual data points or groups of data points. Different strategies can be used to decide on the distance between groups of data points. Single linkage uses the minimum distance between a data point in one group and a data point in another group, complete linkage is the maximum distance, average distance is the average distance between every data point in one group and every data point in another group. Alternatively the centroid of each group can be used to compare groups. This method is not very robust to noise, small changes in the data could make large changes to the structure of the cluster tree. It is also computationally expensive as it needs to calculate pairwise distances between all data points. This was first applied to gene expression data in Eisen et al. (1998).
3.3 Clustering Techniques

Figure 3.7: Example of 3 clusters. (a) K-means algorithm: K-means represents clusters as a single point (cluster centroid). Data points are then allocated to the cluster with the closest centroid. Red circles indicate the cluster centroids. (b) The linking of groups in hierarchical clustering can be represented as a tree structure. In this example there are three large subtrees, corresponding to the three clusters of data points. (c) Mixture model: the dataspase can be represented as a mixture of probability distributions. There are three areas of the data space with high probability, corresponding to the positions of the 3 clusters. Each individual cluster can be represented as a Gaussian distribution with mean and covariance parameters.
3.3 Clustering Techniques

Figure 3.8: Hierarchical clustering: Initially there are 6 groups, each containing a single data point. The distances between groups is calculated. The two groups with the smallest inter-group distance are then joined to form a new larger group. The algorithm continues until a single group containing all data points is formed.

3.3.3 Self-organising Maps

Self-organising maps (Kohonen, 1982) have also been used for microarray data clustering (Tamayo et al., 1999). A set of nodes forming a grid in two-dimensional space are mapped into $D$-dimensional space, initially at random. The nodes are then iteratively adjusted by selecting a point from the data set and moving all the nodes in the direction of this chosen point. The closest node is moved by the greatest distance and those nodes furthest away are adjusted by a smaller factor.

3.3.4 Correlation Graphs

An alternative approach to clustering genes in microarray data is that of correlation graphs (Rougemont and Hingamp, 2003). In this method, nodes in the graphs represented genes and links between nodes represented co-expression of genes. Clusters could then be thought of as dense connected regions of the graph with high curvature. The Pearson correlation distance was calculated between each pair of profiles and a link was drawn between the corresponding genes if the Pearson correlation value was higher than a given threshold. The links in the graph could correspond to both positive and negative correlation. For each node the curvature was calculated and any nodes with a curvature below a selected
threshold value was deleted. This caused the graph to be split into several components, each component could be interpreted as cluster of co-expressed genes.

### 3.3.5 Mixture modelling

A mixture model approach models the distribution of data in terms of a set (mixture) of probability distributions (components). Each data point is modelled as an observation generated from the mixture of distributions. A major advantage of the mixture model approach over methods such as $K$-means clustering is that the probability density functions provide an explicit and principled way of handling uncertainty. For example, the $K$-means method only provides information about the distance of data points from the centre of each cluster. A mixture model incorporates information about the certainty with which data points are associated with each of the mixture components and the overall model of the data.

A Gaussian mixture model (GMM) approach treats each data point as an observation, $x_n (n = 1...N)$, that is generated from a mixture of $K$ multivariate Gaussian distributions. The model is specified by a component set, $\theta = \{\pi, \mu, \beta\}$, in which $\pi = \{\pi_1, ..., \pi_K\}$, $\mu = \{\mu_1, ..., \mu_K\}$, $\beta = \{\beta_1, ..., \beta_K\}$ and where $\pi_k$, $\mu_k$ and $\beta_k$ are the mixing probability, mean vector and precision matrix (inverse covariance) for component $k$ respectively. $L_n$ is the cluster label for the $n$th data point and so $L$ can be thought of a set of hidden variables whose values are also of interest. Figure 3.9 shows a graphical model representation of how data points are assumed to have been generated from a Gaussian mixture model.

A component $k$ (with corresponding cluster label) is chosen according to the mixing probabilities in $\pi$, such that $p(L_n = k|\pi) = \pi_k$. A data point is then generated from a Gaussian distribution with mean $\mu_k$ and precision $\beta_k$. The overall probability that a data point $x_n$ was generated from the model is thus given by:

$$p(x_n) = \sum_{k=1}^{K} p(L_n = k|\pi)p(x_n|\mu_k, \beta_k)$$ (3.11)

Figure 3.7(c) shows how the probability changes at different positions in the data space, with much higher probabilities of data being generated close to the center of the mixture.
3.3 Clustering Techniques

Figure 3.9: Graphical model of a Gaussian mixture model. A component $k$ (with corresponding cluster label) is chosen according to the mixing probabilities in $\pi$, such that $p(L_n = k | \pi) = \pi_k$. A data point is generated from a Gaussian distribution with mean $\mu_k$ and precision $\beta_k$.

Unlike $K$-means clustering, each data point has a probability of association with each of the underlying components, given by the label posterior probabilities, $p(L_n = k | x_n)$. Thus the mixture model can be thought of as a projection of the $D$-dimensional space into a $K$-dimensional space of posterior probabilities for each data point given each component. The positions of the points are constrained to lie in the region of space greater than or equal to zero along all dimensions and such that the total of the positions across all dimensions sum to one. In $K$-means clustering there are also no parameters for covariance, so all clusters are assumed to be spherical and of equal volume, but in a mixture model the different components can have a different shapes, volumes and orientations, determined by the covariance matrices. In order to cluster the data, the cluster labels can be choosen such that $L_n$ takes a value from $1, \ldots, K$ corresponding to the mixture component for which the data $x_n$ has the highest posterior probability, so $L_n = \max_k p(L_n = k | x_n)$.

Inferring The Model Parameters With Expectation Maximisation (Dempster et al., 1977)

The parameters of the mixture model and cluster labels are are not usually known in advance and therefore must be inferred from the data. The EM algorithm can be used to estimate the component parameters by choosing the parameter values that maximum the likelihood. The algorithm involves an iterative update of the parameter estimates and the posterior of the mixture labels until the log likelihood converges within a specified
tolerance. The log likelihood of the entire data set can be computed by summing the log likelihood for all data points:

$$\mathcal{L} = \sum_{n=1}^{N} \log p(x_n)$$

(3.12)

where

$$p(x_n) = \sum_{k=1}^{K} p(L_n = k | \pi)p(x_n | \mu_k, \beta_k) = \sum_{k=1}^{K} \pi_k N(x_n; \mu_k, \beta_k)$$

(3.13)

**E-step:** Infer the posterior probabilities, \( p(k|x_n) \), of the mixture labels given each datum. For each component \( k = 1 \ldots K \) we have

$$p(k|x_n) = p(L_n = k | x_n) = \frac{\pi_k N(x_n; \mu_k, \beta_k)}{p(x_n)}$$

(3.14)

**M-step:** re-estimate the parameters for each component in the mixture model given the updated E-step posteriors. The following set of update equations are used:

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} p(k|x_n), \quad \mu_k = \frac{\sum_{n=1}^{N} x_n p(k|x_n)}{\sum_{n=1}^{N} p(k|x_n)}$$

and

$$\beta_k^{-1} = \frac{\sum_{n=1}^{N} (x_n - \mu_k)^T (x_n - \mu_k) p(k|x_n)}{\sum_{n=1}^{N} p(k|x_n)}$$

(3.15)

Mixtures of multivariate Gaussians have been used to model the distribution of genes in microarray data sets (Yeung et al., 2001; Ghosh and Chinnaiyan, 2002). They clustered each data set multiple times with different numbers of clusters and covariance matrix parameterisations, then used a Bayesian information criterion (BIC) score to find the best model for the data. Initialising the parameters, in particular the covariance matrices can have a profound effect on the clustering result of this algorithm and it is therefore important that initialisation is carefully considered. This is because the EM algorithm will converge to a local maximum solution for the log likelihood, not necessarily the global maximum, and the initialisation affects which local maximum will be chosen.
Model order selection: To find the optimal number of clusters, different mixture models are initialised that contain different numbers of components and then the Bayesian information criteria (BIC; Schwarz, 1978) can be used to calculate the optimal model order. The BIC score is given in equation 3.16, where $\mathcal{L}$ is the log likelihood of the model, $v$ is the number of parameters in the model and $N$ is the number of data points being modelled. The score, which is equivalent to the minimum description length, finds a balance between the log likelihood, which will always increase as more components are added, and overfitting to the data by using more components than necessary.

$$BIC = 2\mathcal{L} - v\log(N)$$

(3.16)

Bayesian Mixture Modelling

Although an EM-BIC approach has some justification, it remains a penalised maximum-likelihood method and so does not have the principled perspective of fully probabilistic (Bayesian) models which take into account uncertainty at all levels of inference. So there may be multiple models (sets of parameter values) that can explain the distribution of data in the data set, the EM approach chooses one model, the one that maximises the likelihood for a given number of components, and then uses a penalty term to select the optimal complexity. A Bayesian model, models not only the uncertainty in the data but also the uncertainty in each of the component parameters. So the component parameters are not point estimates, but can take a range of values, and are thus modelled as a variable described by a probability distribution with associated hyperparameters. Figure 3.10 shows a graphical model describing Bayesian mixture modelling and shows the relationship between the data and the prior distributions of the component parameters.

Bayesian modelling involves finding the posterior distribution of the parameters and hidden variables given the data, $p(\Theta|X)$, where $\Theta = \{\theta, L\}$. This posterior distribution is therefore a probability density function over all possible values of the parameters and cluster labels. However, in order to find the true posterior, marginalisation over all possible sets of parameters is required to calculate the evidence, $p(X)$. As this is analytically
3.3 Clustering Techniques

Figure 3.10: Graphical model of data generated from a Bayesian Gaussian mixture model. There is uncertainty in the component parameters as well as the data, therefore values for the component parameters are first drawn from probability distributions according to the hyperparameters and then the data point is generated from the model according to these parameter values.

intractable for mixture models the true posterior cannot be found but methods exist that provide good approximations.

Some sampling strategies have been used with the Bayesian analysis of microarray data, to approximate the model posterior. For example, Vogl et al. (2005) sampled from the posterior of all model parameters using a reversible jump Markov chain Monte Carlo method and Medvedovic et al. (2004) used a Gibbs sampler to sample from the posterior of cluster labels and then used model averaging to infer which genes should be allocated to the same cluster label. Although these sampling strategies can provide good results, sampling from the model posterior has the inherent drawback that it is computationally inefficient, especially for large data sets such as those encountered in the bioinformatics field. A more efficient alternative to sampling has been provided by work in variational Bayesian inference (MacKay, 1995; Attias, 1999; Choudrey et al., 2000). Using a variational approach, approximate solutions to such problems can be obtained that retain the desirable properties of the models, such as full incorporation of parameter uncertainty and natural model selection.

The variational Bayes (VB) framework has already been applied to mixture modelling of gene expression data (Teschendorff et al., 2005; Muro et al., 2003) and shown to provide a more robust and efficient model order selection process than alternative methods. The
variational Bayes approach was compared to the EM+BIC using simulated data (Teschedorff et al., 2005), the results showed that the misclassification rate and the cluster-number prediction error was lower in the variational Bayes approach. Therefore variational Bayes can be better at modelling the underlying structure than the EM approach.

**Variational Bayes** Defining $q(\Theta | X)$ to be a tractable approximation to the true posterior $p(\Theta | X)$ it is possible to express the log evidence in terms of the negative variational free energy, $F[q(\Theta | X)]$, and the Kullback-Liebler divergence between the true and approximating posterior (equation 3.17). The Kullback-Liebler divergence measures the difference between two distributions, equalling zero if the approximating posterior perfectly equals the true posterior. Therefore, we wish to minimise this divergence but unfortunately as the true posterior is unknown this divergence cannot be calculated in practice. However, as $\log p(X)$ is constant with respect to $\Theta$, maximising the negative free energy is equivalent to minimising the KL-divergence, and this can be done.

$$
\log p(X) = \log \frac{p(X, \Theta)}{p(\Theta | X)} = \int_\Theta q(\Theta | X) \log \frac{q(\Theta | X)p(X, \Theta)}{q(\Theta | X)p(\Theta | X)} \, d\Theta = \int_\Theta q(\Theta | X) \log \frac{p(X, \Theta)}{q(\Theta | X)} \, d\Theta + \int_\Theta q(\Theta | X) \log \frac{q(\Theta | X)}{p(\Theta | X)} \, d\Theta = F[q(\Theta | X)] + KL[q(\Theta | X) || p(\Theta | X)]
$$ (3.17)

A mean-field variational approach assumes that the prior and approximating posterior are separable such that $p(\Theta) = \prod_j p_j(\Theta_j)$ and $q(\Theta | X) = \prod_j q_j(\Theta_j | X)$. In the case of a Gaussian mixture model this implies the following separations:

$$
p(\Theta) = p(L|\pi)p(\pi) \prod_k p(\mu_k)p(\beta_k)
$$

$$
q(\Theta | X) = q(L|\pi, X)q(\pi | X) \prod_k q(\mu_k | X)q(\beta_k | X)
$$

(3.18)

Distributions then need to be choosen to model the component parameters. Penny (2001) showed how a joint Normal-Wishart distribution ($\mathcal{N} - \mathcal{W}$) can be used to model the component means and covariances and a Dirichlet ($\mathcal{D}$) for the mixing probabilities. The variables
\( \alpha_k, B_k, \tau_k, m_k \) and \( \lambda \) form the set of model hyperparameters which need to be optimised in order to maximise the negative free energy, such that:

\[
q(\beta_k | X) = \mathcal{W}(\alpha_k, B_k) \\
q(\mu_k | \beta_k, X) = \mathcal{N}(m_k, \tau_k \beta_k) \\
q(\pi | X) = \mathcal{D}(\lambda) \tag{3.19}
\]

The same prior is used for each of the model components, in which \( \alpha_0, B_0, \tau_0, m_0 \) and \( \lambda_0 \) are the hyperparameters for the prior distributions such that:

\[
p(\beta_k) = \mathcal{W}(\alpha_0, B_0) \\
p(\mu_k | \beta_k) = \mathcal{N}(m_0, \tau_0 \beta_k) \\
q(\pi) = \mathcal{D}(\lambda_0 I_K) \tag{3.20}
\]

The variational Bayes approach can be implemented in an EM-like procedure, in which the label posterior is calculated in the E-step and then a set of coupled update equations (Penny, 2001) are used to provide new estimates for the hyperparameters in the M-step. The iteration continues until the negative free energy converges. A suitable choice for the prior hyperparameters and an initialisation of the posterior hyperparameters are required for the algorithm to be implemented.

**Model-order selection** As maximising the variational negative free energy is equivalent to minimising the divergence between the true and approximating posterior, model-order selection could be performed by evaluating models of increasing complexity and choosing that with the maximum value for \( F[\Theta|X] \). However, this is actually not necessary as the free energy maximisation naturally penalises extraneous complexity. Components of the model which do not contribute to the explanation of the data will converge to \( \pi_k = 0 \) and hence effectively remove them from the model. Therefore, unlike the EM algorithm, in which multiple projections are required, with different values of \( K \) and the models compared, only one projection has in required with the variational Bayes approach provided that it is initialised with sufficiently high complexity (figure 3.11). This penalty
can be seen by decomposing the negative free energy term into the average log likelihood and KL-divergence between the approximating posterior and the prior (equation 3.21). Thus, to maximise the negative free energy, the posteriors will converge towards the prior distribution \( p(\Theta) \) in order to minimise the KL-divergence, and the mixing probabilities will converge to 0, unless the mixture component provides a sufficiently increase in the average log likelihood.

\[
F[q(\Theta|X)] = \int_{\Theta} q(\Theta|X) \log \frac{p(X, \Theta)}{q(\Theta|X)} d\Theta \\
= \int_{\Theta} q(\Theta|X) \log \frac{p(X|\Theta)p(\Theta)}{q(\Theta|X)} d\Theta \\
= \int_{\Theta} q(\Theta|X) \log p(X|\Theta) d\Theta - \int_{\Theta} q(\Theta|X) \log \frac{q(\Theta|X)}{p(\Theta)} d\Theta \\
= \langle \log p(X|\Theta) \rangle_{q(\Theta|X)} - KL[q(\Theta|X)||p(\Theta)]
\]
Figure 3.11: Model complexity on toy data containing 3 clusters. Models learnt using EM and VB algorithms when initialised with 3 and 20 components. Red dots and circles represent the mean and covariance parameters for the EM models. Magenta dots and circles represent the expected value of the mean and covariance parameters for the VB models. Cyan dots and circles represent samples taken from the posterior distributions of the means and covariances for the VB model, and thus indicate the uncertainty in the model parameters. (a) The EM algorithm finds a good model for the data (b) the EM algorithm fits all 20 mixture components to the data as the log likelihood will always increase with the number of components (c) VB fits a good model to the data (d) VB naturally penalises complexity and removes all but 3 of the components by setting their posterior mixing probability to zero.
3.3.6 Times Series Data Analysis

Other models that have been used for clustering gene expression data, mainly time series data, include Bayesian decomposition (Moloshok et al., 2002); hidden Markov models (Schliep et al., 2003; Ji et al., 2003); spline curves (Bar-Joseph et al., 2002); autoregressive curves (Ramoni et al., 2002); piecewise linear functions (Filkov et al., 2002) and a shaped based similarity measure (Balasubramaniyan et al., 2005).

3.4 Data Transformations

Although the choice of technique is critical in any unsupervised data analysis, the choice of metric which defines similarity between points in the data space is also important. In Gaussian mixture modelling approaches, local metrics are inferred (via the covariance estimates) which take into account the local distribution of data. Global transformations of the data also affect the relationships between data points and hence will give rise to different cluster solutions. Yeung et al. (2001) tried different transformations such as square root and standardisation of the data prior to the clustering so that the genes were clustered on the shape of the expression profile rather than the absolute expression values. An example of when clustering on the shape of the profile rather than just the intensity changes is given in figure 3.12. In this example there are three different expression patterns, but because the patterns have different intensity changes, a Gaussian mixture model of the intensity change data finds 9 clusters. By standardising the data first, the 3 clusters are successfully found by a mixture model approach. However, it is not always easy to interpret the expression patterns associated with the clusters, and how the clusters differ from standardised data as standardisation transforms data to a scale that is not meaningful in terms of expression change.

3.5 Summary

Projection techniques can be used to transform data from some matrix $X$ of dimensionality $D$ into a different representation $Y$ of dimensionality $d$. Patterns can also be automatically
Figure 3.12: Data transformations - (a) 3 groups of genes showing similar shaped expression profiles but varying intensity changes (b) variational Bayes mixture modelling does not distinguish three clusters as they have such a large variation in intensity change (c) standardising the data groups the profiles with similar shapes (d) a variational Bayes mixture model of the standardised data successful locates the different shaped clusters. Therefore, transforming the data prior affects the distances between data points and the results of data projections. Although in this example the different clusters have been distinguished it is not always easy to interpret the expression profile associated with a cluster, or to understand the differences in expression profiles that distinguish different clusters. This is because the standardisation projects data on to axis that are not meaningful in terms of expression change.
extracted from data by using clustering techniques. These project data either on to a tree
structure defined by a set of nodes and links between them or into a $K$-dimensional space
(so $d = K$ for these techniques) corresponding to a number of underlying components.
The models differ in their parameterisations, the way that they model the groupings of
data points and their model order selection. Variational Bayes mixture modelling offers
a fully probabilistic approach that provides a principled model order selection as well as
computationally efficiency. In the analysis of gene expression data it is also important to be
clear about what expression patterns are of interest, and what it means to be co-regulated
for the purpose of the particular study. For instance, it may of interest to find genes that
share similar intensity changes. Alternatively, it may be of interest to find all genes that
show the same pattern of change but the magnitude of intensity may be different. Data
transforms can be applied to the data to project the data into a space where the distances
between data points have an alternative definition.
The techniques reviewed have had some success at locating patterns hidden within gene
expression data sets. However, the results of the techniques can still be difficult to interpret,
for instance it can be difficult to understand what underlying pattern a cluster cor-
responds to, particularly for large clusters and after the application of certain data trans-
forms. Ideally, the data projections should project the data to a representation from which
information about the expression changes can be easily inferred.
Current methods of visualising gene expression data have mainly focused on plotting the
expression patterns for different clusters or using a heatmap. However, for large data sets
or large clusters it can be difficult to visualise the similarities and differences between
genes within the cluster and as well as inter cluster separation. Heatmaps can become
very large and difficult to interpret, with the squares representing each gene becoming
very small. It is also difficult to visualise extra information about the genes with these
visualisations and is not easy to put a biological meaning to pattern associated with each
cluster, particularly if a shaped based distance measure is used. Dimensionality reduction
techniques could be used to visualise data on a 2D map, by reducing the data to represen-
tation where $d = 2$. Three different techniques have been reviewed in this chapter: PCA; LLE and GTM. These make different assumptions about the features of the data that are important to retain in the lower dimensional representation.
4

Visualisation of Simple Data sets & Structural Analysis of Gene Expression Data

4.1 Introduction

The previous chapter discussed how visualising gene expression data on a 2D expression map could be advantageous and introduced some dimensionality reduction techniques that could be used to project the data into a 2D representation. To reduce the dimensionality of a data set to 2D requires assumptions to be made about the data, for example in the form of a basis that describes the variability in the data or an assumption that the data points are distributed along the surface of a 2D manifold. In this chapter, Principal component analysis (PCA), Locally Linear Embedding (LLE) and Generative Topographic Mapping (GTM) are applied to some simple dimensionality reduction problems to observe the effect that (a) the data set, (b) the distribution/manifold of the data (c) the algorithm parameters have on the dimensionality reduction and to demonstrate the sort of structures that the different methods highlight. It shows that each dimensionality reduction technique works well for visualising certain types of data structures but that they can provide poor results if applied to data that does not have these structures. This chapter then attempts to find the structures hidden in some gene expression data sets in order to decide which technique may be the most appropriate for visualisation of that type of data.
4.2 Simple Data sets

4.2.1 Linear Problem

In this problem 100 datapoints describe a linear relationship in two dimensions. The algorithms were programmed to reduce the data to one dimension. As expected PCA was able to reduce this linear data set with good results. Figure 4.1 shows the original data set and the PCA reduction into 1D space.

LLE was used with a range of values for the parameter $K$, which specifies the number of nearest neighbours to use. Figure 4.2 shows that with $K = 2$ or 5 the relationship of the data was not preserved but good results were obtained with $K = 10$. Therefore even on the simplest of problems the value of $K$ used is critical in providing a good result from the LLE algorithm.

GTM gave good results on this problem for all parameter values tested. Two parameter values were changed, $K$, the number of grid points in the latent space and $M$, the number of basis functions. Figure 4.3 shows the GTM mapping of the datapoints. Therefore this linear problem could be reduced successfully with all three methods, however LLE was the most sensitive to parameter values.

---

Figure 4.1: PCA results for the linear data set. (a) The data points with the principal components displayed. (b) The data is then rotated so the principal components form the axes. (c) The dimensionality reduced to 1D by only using the first principal component. The dimensionality reduction has correctly preserved the ordering of the data points.
Figure 4.2: The 1D mapping of the linear data calculated with the LLE algorithm. The value of K is critical in giving a good mapping of the data, but when a suitable value is chosen the results are very good.

Figure 4.3: The GTM results for the linear data set with different parameter values. GTM gave good results with all parameter values used.
4.2 Simple Data sets

4.2.2 S-curve Manifolds

A series of data samples were generated for the s-curve manifolds. Forty data sets were produced, consisting of ten random samples from the manifold of sizes 500, 1000, 2000 and 4000. The different sample sizes were used in order to examine the effect that the size of the data set had on the results. It was expected that the greater the number of datapoints used the better quality results produced, as the manifold would be better described if more data points were used. More than one sample was taken for each sample size to examine the effect that the actual set of data points had on the result.

The PCA method was consistent across samples of the same size and of similar shape and mapping with differing number of data points. However, none of the results correctly found the spread of datapoints across the manifold. This is because PCA can only find linear components. The PCA can view the s-curve from different angles but can not unfold the manifold. Some typical results of PCA on different data sets can be seen in figure 4.4.

The LLE method was run on the data sets with a range of values for $K$. The mapping varied dramatically between data sets with 500 points (see figure 4.5). The best results were found when $K = 10$ but the quality was very variable between data sets. It is therefore important that LLE is used with a data set that samples the manifold adequately as the method is extremely sensitive to the value of $K$ and the actual points with small sample sizes.
Figure 4.5: The LLE mapping for five data sets, all with 500 data points. The best results were given when $K = 10$ (middle row) but the results were not consistent, showing that LLE does not work well will small sample sizes and the value of $K$ can have dramatic effects. Both $K$ being too small and too large gives bad results.
4.2 Simple Data sets

Figure 4.6: The LLE mappings for larger data set with $K = 10$. Top row is results for data sets with 1000 points and bottom row show results for data sets with 2000 points. With more data points the LLE algorithm can give good results of consistent quality.

With 1000 datapoints LLE managed to unfold the manifold correctly when $K = 10$ or 20 and with 2000 datapoints $K = 10, 20$ or 40 worked well (see figure 4.6). Therefore increasing the number of data points gives more consistent results and allows the method not to be so sensitive to the value of $K$, although values too small and too large will still produce poor results.

Using $K = 5$ gave poor results on all sizes of data set tested, local sections of the manifold seemed to be grouped together but the global picture of the manifold was lost because not enough neighbours were considered for each point. Increasing the number of nearest neighbours too far can however mean that points are considered to be on locally linear patches of the manifold when actually they are not and trying to preserve these incorrect relationships causes the mapping to correctly merge areas of the manifold in lower dimensional space.

The GTM method was run on the s-curve data sets with varying values for $K$ (number of latent variables) and $M$ (number of basis functions). Increasing $K$ improved the quality of the results and it seemed to work best when $K > M$. Again the results were of similar quality across the data sets of the same sample size but the quality improved as the sample size increased. Like the LLE algorithm, the parameter values had a dramatic effect on the
4.2 Simple Data sets

Figure 4.7: The GTM reductions of a data set with 2000 points using different parameter values. Small values of $K$ were bad because the grid in latent space was not fine enough to place the datapoints, therefore projected data clustered together around certain points rather than being smoothly spread. Unlike LLE making the value of $K$ larger did not seem to make the results worse, therefore the parameter values do not seem to be as critical as LLE provide the grid size is quite large. Some GTM results can be seen in figure 4.7.

4.2.3 Swiss Roll Manifolds

A series of data samples were also generated for the swiss roll manifold in a similar way to the s-curve samples. PCA gave better results on the swiss roll than the s-curve, as it rotated the swiss roll to a view that allowed the different parts of the swiss roll to be seen. However it just looked from a side view rather than unravelling the manifold, so for more complicated manifolds PCA would not be useful. It also did not work on all of the data sets, so reasonable results could not be guaranteed. Some of the PCA results can be see in figure 4.8.

The swiss roll could not be correctly mapped to a lower dimension using the LLE algorithm with only 500 data points. With 1000 data points some good results were obtained with $K = 10$, the only consistently good results found with the parameters tested was using $K = 20$ on data sets of 2000 points. Some of the LLE results for the swiss roll data sets can be see in figure 4.9.
4.2 Simple Data sets

The GTM algorithm did not seem to produce a good mapping of the swiss roll manifold into 2D space with the tested parameters and data sets. An example of the manifolds for one of the data sets can be seen in figure 4.10. The problems may have occurred because this implementation of GTM initialised the mapping with PCA. Therefore LLE and GTM produced worse results on the swiss roll than the s-curve manifold but the LLE can successfully find the manifold provided there are sufficient data points and $K$ is within the optimum range. GTM should also be used with caution as the swiss roll data sets highlight that there are fairly simple manifolds that the GTM can not be used successfully with.

Figure 4.8: PCA results for different data sets of 500 data points from the swiss roll manifold. The results are very variable for the different data sets, although some show the swiss roll from a side view and therefore do correctly separate the different areas of the manifold this result can not be guaranteed.
4.2 Simple Data sets

Figure 4.9: LLE results for different swiss roll data sets with $K = 20$. The top row shows the LLE mapping for 5 different data sets of 500 points. LLE failed to give good results on this manifold with only 500 data points. Only one of the five data sets shown with 1000 points (middle row) gave good results for the swiss roll manifold. This shows the same inconsistency and sensitivity to data set size that was seen with the s-curve. However the bottom row of results show LLE mappings for data sets with 2000 points, and with that data set size LLE does give good results for the swiss roll manifold.
Figure 4.10: A typical set of GTM results for a swiss roll data set of 2000 points. Even with 2000 points GTM did not produce good results for this manifold, therefore although GTM was very useful for the linear and s-curve problems, it must be used with care as there are manifolds it does not work well with.
4.3 Visualisation of Clustered Data

The previous section demonstrated how GTM can be used to locate a 2D manifold in a data set of higher dimensions. By altering the parameters, the smoothness of the manifold can be adjusted. Visualising the distribution of the data along this manifold can be achieved by unravelling the manifold into 2D co-ordinates and associating each datapoint with a position in these 2D co-ordinates, based on the datapoints position on the manifold. The reduction in dimensions is therefore achieved by exploiting this 2D structure. However, for data sets in which there are distinct clusters of data points rather than data spread on a continuous 2D manifold the visualisation task should perhaps be to highlight these separate groups of points and the relative distances between groups, rather than the position of data on a 2D structure. The choice of the number of basis functions and latent variables can be important when attempting to visualise the separate groupings in clustered data.

The number and width of the basis functions is important, as overfitting can occur if the number of basis functions is too large and the manifold is allowed to become too complex. When a small number of basis functions is chosen, the relative positions of the Gaussian distributions in the data space are constrained, leading to a much smoother manifold, and therefore the distributions model the space between clusters as well as the groups of data points. Increasing the number of basis functions, allows much greater flexibility in the mapping of the latent variables in to the dataspace, allowing a much more complex manifold, in which the majority Gaussians model data points.

4.3.1 1D Example

Figure 4.11 shows a simple data set containing 3 clusters of data points. In this example a 1D GTM manifold is fitted to the data and the data points can then be visualised according to position on the 1D manifold. From this example, it can be seen that the smaller number of basis functions captures a smooth transition between the clusters, and will allow 3 distinct clusters to be seen in a 1D visualisation. Taking the hypothesis that the underlying structure in the data is 3 clusters of data points linked by a 1D manifold then
4.3 Visualisation of Clustered Data

Figure 4.11: 1D GTM example: 2D Data set containing three clusters of data, modelled using a 1D GTM manifolds. Different numbers of basis functions and latent variables were used. Green circles show the positions of the Gaussian distributions in the data space. Small numbers of basis functions constrain the positions of the Gaussians to a much smoother manifold. More stretching and compressing of the manifold can occur with a larger choice of basis functions, allowing the manifold to overfit the data. Increasing the number of latent variables does not have a dramatic effect on the shape of the manifold but will allow better resolution and mapping of inter-cluster space in a 1D visualisation of the data.

the visualisation using GTM to reduce the dimensionality will provide useful information about the groups within the data and their distribution along the manifold, although some information regarding the distribution of data within each cluster will be lost. However, if the clusters are simply separate groups of data points rather than linked by an underlying manifold, the projection of points in 1D will not preserve all the inter-cluster distances. For instance, two of the clusters are much closer in the data space than would be suggested by their positioning along the manifold. Therefore, if GTM is used to visualise clustered data, rather than data with an underlying manifold, useful information about different groupings may be seen, but the relative distances may not be correctly represented.
4.3 Visualisation of Clustered Data

4.3.2 2D Example

This data set contains five clusters in 3D space. The visualisation of the data set using a 2D GTM manifold with various choices of the number of basis functions can be seen in figure 4.12. Although the inter-cluster separation can not be reliably visualised, the different groups of data points can be clearly seen in figure 4.12. With the larger choices of basis functions, the relative positions of the Gaussian distributions on the manifold can be much more flexible, allowing a lot more stretching and compressing of the manifold and consequently all the Gaussian distributions can be mapped around the clusters of data rather than between clusters. This has the effect that the clusters are not as distinct in the 2D visualisation. Figure 4.13 shows the corresponding magnification factors and 3D manifolds.

Figure 4.12: 2D GTM Example: Five clusters in 3D space, visualised with a mapping to 2D using GTM. The different visualisations use different numbers of basis functions (a)4, (b)16, (c)25, (d)36, (e)64, (f) 81. All use 3600 latent variables. The clusters are more distinct with a smaller number of basis functions.
Figure 4.13: 2D GTM Example: The magnification factors (a-e) and manifolds (f-g) for the GTM models with 4, 16, 25, 64 and 81 basis functions. The magnification factors can be used to help separate the clusters. The light areas show small magnification factors, were the manifold is compressed around a high density of data and the darker areas show the stretching of the manifold, where there is a much lower density of data. The manifolds show the increasing complexity and overfitting to the data as the number of basis functions is increased.

Comparison With Other Methods

For comparison PCA and LLE were also applied to the clustered data set. The 2D visualisation using PCA does manage to highlight the separate clusters (figure 4.14). However, the principal components may not necessarily give a good separation of the clusters for all data sets, particularly those with higher dimensions. Figure 4.15 shows mappings of the data into 2D space using LLE. The results were very variable, although a reasonable visualisation was seen when \( K = 130 \), other choices for the number of nearest neighbours to consider, did not give such good results.

4.4 Structural Analysis of Gene Expression Data

This section attempts to find the structures hidden in some gene expression data sets in order to decide which technique may be the most appropriate for visualisation of the data. The data sets under investigation are those which include multiple experimental conditions (e.g. time points and/or treatments) and in which large numbers of genes are expected to show changes between conditions. The conclusions of this structural analysis may not
4.4 Structural Analysis of Gene Expression Data

Figure 4.14: PCA visualisation of the clustered data. The PCA result for this data set does successfully highlight the clusters of data. However the principal components will not necessarily provide such a good representation of the data for all data sets. PCA will only provide good visualisations for data sets in which the 2D plane describing the basis of maximum variability gives a good separation of the clusters.

Figure 4.15: LLE visualisation of the clustered data for different values of $K$ (number of nearest neighbours to use), (a)50, (b)100, (c)130, (d)150, (e)300. Each of the five clusters contained 100 datapoints. With $K < 100$, the results were very poor, as only the neighbours within the same cluster were considered, so no information about inter-cluster distances were modelled. A reasonable separation of the clusters was seen when $K = 130$, but the groupings of the data were not as easily visualised with other parameterisations of the LLE algorithm.
Figure 4.16: Comparisons of expression ratios for pairs of conditions. (a)-(d) are from the rice blast data set: (a) hydrophobic surface 7h vs 12h (b) hydrophilic surface 7h vs 12h (c) Hydrophobic 7h vs Hydrophilic 7h (d) Hydrophobic 12h vs hydrophillic 12h. Comparisons (e)-(h) are from the yeast environmental shock data set: (e) Heat Shock 20min vs 30min (f) Hydrogen peroxide 10min vs 20min (g) Hydrogen peroxide 20min vs 30min (h) Heat shock 20min vs Hydrogen peroxide 20min. All show a large proportion of data points close to the origin, and then outlying data at different angles from the origin. The jets of data in (e) show genes that are up-regulated or down-regulated by the similar amounts in at both 20min and 30min following heat shock. However, (f) does not show such clear jet structures, therefore there are many genes that show expression change following hydrogen peroxide shock, there is no clear patterns of change between 10 and 20mins after shock.

be as applicable to gene expression data sets of gene knockout studies, where only a few genes may show a change in expression after the knockout and those in which the aim of the analysis attempts to find differentially expressed between multiple classes.

4.4.1 Pairwise Analysis of Conditions

In order to visually investigate the structures in the microarray data, two-dimensional plots were made by selecting pairs of microarray conditions and plotting the gene expression for those two conditions. Figure 4.16 shows a few of the comparisons of conditions within some microarray data sets. The following observations were made about the microarray data sets, these observations are also illustrated in figure 4.17:

1. A large proportion of data points are found in a circle around the origin. These correspond to genes that show no conclusive regulatory change in either array. This circle
Figure 4.17: Example of the angular dispersion of log fold change data on an example from the rice blast data set. This pair-wise comparison shows the expression data for 12 hours after the start of development on a hydrophobic surface and 12 hours on a hydrophilic surface. The fungus is known to develop very differently on the two surface types and therefore is of interest to find genes that are associated with an increase in expression on the hydrophobic surface but not the hydrophillic surface. (a) The vast majority of genes are close to the centre, with 78% of the 13,666 showing less than a 2-fold change in all conditions of the data set. Those that show larger changes in expression seem to be either up-regulated by similar amounts, down regulated by similar amounts, up-regulated in hydrophobic with little change in hydrophillic or down in hydrophillic but no change in hydrophobic. (b) Shows an angular histogram of the dispersion of genes (c) shows the angular dispersion for genes with greater than 2 fold change in expression.
of data points can therefore be thought of a cluster of uninteresting genes.

2. Some genes lie along or close to a 45 (or 225) degree angle, these correspond to genes which show similar amounts of up-regulation (or down-regulation) between conditions. These genes maybe of interest in the sense that they show similar up- or down regulation in both conditions.

3. All other genes correspond to genes that show some significant amount of up- or down-regulation but are differentially expressed between the conditions. Some genes seem to spread out from the origin towards extreme points in jets of data. The jets occur when a large number of genes show the same regulation patterns but have varying magnitudes. As many genes show the same regulatory pattern and form these jet structures in the data space, these expression patterns could be biologically significant.

By making pairwise comparisons, the differences in expression between the two surface types can be seen (figure 4.18). In the comparison of the two time points on the hydrophilic surface, there is a group genes showing similar levels of up-regulation from the spore sample, and many showing similar levels of down-regulation but not many are differentially expressed between 7 hours and 12 hours. This corresponds to the idea that the fungus is in a similar state of germination at both timepoints on the hydrophilic surface. However, the comparison of the hydrophobic surface time points shows an entirely different story, with much more variation in the up-regulation of genes between the two timepoints, corresponding to the difference in expression associated with appressorium development and with penetration peg emergence.

Visualisation of the pairwise comparisons can therefore allow these general patterns to be seen, but it would be more useful to develop methods that allow patterns to be located across multiple comparisons of conditions. Imagining a D-dimensional space with genes positioned according to their regulatory changes across the $D$ arrays then the data space will contain a D-dimensional hypersphere around the origin. This hypersphere will contain
4.4 Structural Analysis of Gene Expression Data

Figure 4.18: Rice blast data - dispersion of data for pairwise comparisons on the different surface types. (a) The comparison of the two time points on the hydrophilic surface, there is a group genes showing similar levels of up-regulation from the spore sample, and many showing similar levels of down-regulation but not many are differentially expressed between 7 hours and 12 hours. This corresponds to the idea that the fungus is in a similar state of germination at both timepoints on the hydrophilic surface. (b) In contrast the comparison of the hydrophobic surface time points shows much more variation in the up-regulation of genes between the two timepoints, corresponding to the difference in expression associated with appressorium development and with penetration peg emergence. (c) angular histogram of the pairwise comparison on a hydrophilic surface (d) angular histogram for the pairwise comparison in the hydrophobic surface for genes with greater than a 2-fold change in expression.
genes that do not show a significant regulatory change in any of the experiments. All genes that do show potentially interesting regulatory changes will appear as outliers or in the jets of $D$ or fewer dimensions. Points in the jet closest to the origin are the genes that show smaller regulatory changes, those furthest away show the large regulatory changes. Visualising genes in terms of their association with these underlying jets would therefore be useful. PCA and other linear techniques are unlikely to be useful at visualising these jets unless the data contains a single jet structure or if the jets are orthogonal. The next section explores how GTM models the data, and how this affects the visualisation of data using the GTM algorithm to reduce the data to 2D.

4.4.2 Application of GTM to the microarray data

The GTM methodology was used to investigate the hypothesis of angular structures in higher dimensions by looking at the manifolds and magnification factors.

Manifolds in 3D subsets of the data

To illustrate the folding of manifolds in the dataspace, subsets of microarray data containing only three arrays were mapped onto a 2D manifold using the GTM algorithm. To observe the underlying structures in the data both GTM reductions with small number and large number of basis functions were examined. With a small number of basis functions, the general trends in the data should be seen, but the manifold will still be quite smooth. A large number of basis functions will overfit to the data, and the more detailed structure in the data can be seen. Some examples are shown in 4.19. All the manifolds are relatively compressed at the centre around the origin, and very stretched out at the corners to fit to the outlying data. The stretched areas of the manifold are the areas that model the interesting genes.

Magnification Factors

Some examples of magnification factors are shown in figure 4.20. The s-curve manifold showed only very small magnification factors, corresponding to the fact that the manifold
Figure 4.19: GTM manifolds for 3D subsets of the data. (a) rice blast data - hydrophobic conditions (b) rice blast data - hydrophillic conditions (c) yeast data - 3 heat shock time points. The top row shows smoother manifolds with only 16 basis functions, the bottom row shows the GTM manifolds overfitted to the same data sets with 400 basis functions. The complex crumpling and stretching of the manifold is allowed with a large number of basis functions and the distribution of data as angular jets can be seen.
Figure 4.20: The s-curve manifold (a) shows only very small magnification factors, corresponding to the fact that the manifold did not need to stretch a great deal to fit the data. However, the microarray data examples (b) and (c) show small magnification factors for the vast majority of the manifold, but then extremely high magnification factors on the corners and edges of the manifold. 

Removal of the Central Hypersphere

Removing the central hypersphere of data points allowed the manifold to model the ‘interesting’ genes rather than being dominated by the ‘uninteresting’ genes. This idea was investigated, and the complex manifolds and stretching could still be seen (figure 4.21) as the underlying jets of data were still present.

It is still questionable as to whether the data lied on a 2D manifold even if the manifold can take an extremely complex structure and when the manifold is unravelled into 2D, it is difficult to interpret what different areas of the plot mean in terms of expression change.
4.5 Summary

The different techniques have been demonstrated on some simple data sets. Each technique work well for certain types of data structures, or for visualising certain aspects of the data. However, poor results are often obtained on data sets that do not have the op-
timal structure or when a suboptimal parameterisation for the algorithm is used. The initialisation of the algorithm can also affect the results, for instance GTM may provide better results for the swiss roll data set if initialised with a different strategy. Although the parameters do affect the visualisation results for the GTM algorithm, the results are less sensitive to the particular parameters than LLE. GTM can also model non-linear data, unlike PCA. The magnification factors can also provide further information about the distribution of data. GTM makes the assumption of an underlying manifold but was tested on data that in the form of discrete clusters, rather than data spread on a continuous manifold. GTM can cope much better with clustered data than PCA or LLE, but a good choice for the number of basis functions or use of magnification factors are required to aid the visualisation of inter-cluster separation. With an appropriate choice of for the number of basis functions, separation between clusters can be seen, however, the inter-cluster separation might not be accurately modelled between all groups, particularly those on the edges of the manifold.

The structures in the microarray data sets were investigated to establish which models may be most appropriate for the visualisation of microarray data. By taking pairwise comparisons of the data for various conditions, a gene can be viewed as a datapoint positioned in 2D space, its angular dispersion from the centre describing a relative change in expression between the conditions and the radial dispersion from the centre providing information about the magnitudes of expression changes. If many genes show similar relative changes across a series of conditions, then these genes will be distributed in a ‘jet’ of data (angular structure) in multi-dimensional space. On the plots showing pairwise comparisons of conditions, these angular structures can be seen. Examination of of GTM manifolds and magnification factors for data in a larger number of comparisons also suggests that these angular structures exist, with large percentage of genes contained within a central hypersphere of data showing little regulatory change in any condition. Applying GTM directly to these data sets therefore may not highlight these interesting structures. Instead, it may be advantageous to firstly remove the large number of genes that show
little changes in expression and then model the genes in terms of their association with
the angular patterns present in the data.
Chapter 5 develops a technique that will decompose the data into these angular structures.
This decomposition is equivalent to the transformation of the data into an angular repre-
sentation followed by clustering of the data based on their angular dispersion over a series
of pairwise comparisons of conditions. Chapter 6 the develops a modified version of the
GTM algorithm that can be used with the transformed data to visualise the angular struc-
tures. The visualisation can be done in conjunction with the decomposition in chapter 5 to
provide information about the patterns associated with different areas of the visualisation
plot.
Angular Distribution Decomposition

5.1 Introduction

This chapter develops a method of decomposing data into clusters based on their distribution in angular space. It does this using two data projections. Firstly, the data is projected into angular space, the axes of which corresponds to a set of pairwise comparisons of conditions. Secondly, a modified mixture model approach is used to model the distribution of data in this angular space, projecting the data into $K$ dimensional space, the axis of which correspond to the posterior probabilities of the data given the $K$ mixture components. The genes can be allocated to clusters based on their position in this posterior probability space, for instance by associating a gene with the component for which it has maximal posterior probability. As variational Bayes mixture modelling offers a fully probabilistic and computationally efficient approach as well as principled model order selection, it is used as the basis of the clustering technique presented in this chapter.

This work follows on from the previous chapter, the results of which suggested decomposition of the data into angular structures may allow underlying patterns to be found as well as aiding the visualisation. How this method can be used in conjunction with GTM to provide more informative visualisation will be discussed in chapter 6.

The methodology also follows on from the discussion in the review chapter on the use of data transforms and clustering techniques by suggesting a way of transforming and modelling which can be much more informative than other transforms when looking for genes showing similar relative changes in expression across the conditions. Modelling data after it has been transformed in this way allows regulatory information about the clusters to be automatically interpreted. The pairwise comparisons of conditions (axis of
the angular space) can be chosen to reflect the purpose of the data analysis, and hence this transformation gives control over the definition of similarity between profiles. The automatic interpretation allows the clusters most relevant to an investigation to be easily located for further investigation. It is also shown how the spread of expression data within a cluster can be observed via two measures, the probability a gene is associated with a cluster, giving a measure of how its expression conforms to the pattern associated with that cluster, and the radial distance, which provides a measure of the magnitude of intensity change. These two measures can therefore be used to identify subgroups or locate genes warranting further experimental investigation.

5.2 Methodology: Data Projections

5.2.1 Projection to Angular Space

The log fold change data, $X$, can be transformed into angular data, $A$, by taking $C$ pairwise comparisons of different conditions. If the $i$th comparison ($i = 1...C$), compares conditions $t$ and $u$ then for the $n$th gene, this can be treated as a Cartesian co-ordinate $(x_{nt}, x_{nu})$ and converted into a polar co-ordinate $(a_{ni}, r_{ni})$ with $a_{ni}$ in the range $[0, 2\pi]$ using equations 5.1 and 5.2. $y_{ni}$ is the ratio for the two conditions calculated as $y_{ni} = \frac{x_{nt}}{x_{nu}}$. It can be seen therefore, that this transformation is essentially a function of the ratio of expression change but transforms data to a representation that can suitably distinguish different patterns. For instance, the ratio transformation cannot distinguish between a comparison in which both log fold changes are positive and one in which both are negative, but these are entirely different expression patterns.

\[
\begin{align*}
    a_{ni} &= \begin{cases} 
    \arctan(y_{ni}) & \text{if } x_{nt} \geq 0, x_{nu} \geq 0 \\
    \arctan(y_{ni}) + 2\pi & \text{if } x_{nt} > 0, x_{nu} < 0 \\
    \arctan(y_{ni}) + \pi & \text{if } x_{nt} < 0 
    \end{cases} 
\end{align*}
\]  
\[ r_{ni} = \sqrt{x_{nt}^2 + x_{nu}^2} \]  

A comparison of the effect of the different transforms on a simple example of yeast heat shock genes following heat shock treatments is shown in figure 5.1. Figure 5.1(a) shows the
log fold change data and the genes share general ‘features’ in the shape of their expression patterns, in the sense that they are up-regulated in response to an increase in temperature and down-regulated in response to a decrease in temperature, but they are not well grouped in the values of their fold change. Standardising the data for gene, by subtracting the mean and dividing by the standard deviation, transforms the data such that genes with similar ‘shaped’ profiles are grouped together (figure 5.1(b)). A mixture model could therefore be applied to this transformed data in order to find co-regulated genes. However, the component parameters from such a model would correspond to the mean and variation of the standardised version of the data and would not have a straight-forward interpretation in terms of gene expression. Figure 5.1(c) shows the similarities in the angular data for the example of the heat shock response genes. When the angular transform is applied the genes responding to heat shock treatments are tightly grouped, and these genes will appear in the same angular structure in the data set. Importantly, the axis of the angular space can also be directly translated in terms of expression change. For instance, an angle of $\frac{\pi}{4}$ radians implies similar up-regulation in both conditions, whereas an angle of $\frac{\pi}{2}$ implies up-regulation in condition 2 but no change in condition 1. This interpretation is discussed in more detail in section 5.3.1. The data projections used in the angular distribution decomposition therefore models the relative changes in gene expression between certain conditions as the ‘features’ of interest when grouping genes and highlighting expression patterns hidden within the data.

5.2.2 Mixture Modelling of Circular Data

The existing mixture modelling approach needs to be adapted to use a mixture of Wrapped Normal distributions rather than Gaussian distributions, as the distance between two angles needs to be calculated as the wrapped distance. For example, the wrapped distance between 5 and 355 degrees is 10 rather than 350 degrees. Thus, the technique presented here is a mixture modelling approach that fits a mixture of wrapped Normal ($WN$) distributions to data after it has been transformed to a set of angular values. The probability density for each observation is modelled as:
Figure 5.1: Example of expression data for eight heat shock response genes in the yeast *Saccharomyces cerevisiae* following heat shock experiments (Gasch et al., 2000). Different transformations of the data affect the similarity between the genes in the data space: (a) log fold change (b) standardised log fold change (c) angular transformation. The expression profiles for these genes share similar ‘features’ but are but more tightly grouped after the data has been transformed in (b) and (c). Conditions 1-8 correspond to different times after an increase in temperature from 25 to 37°C and conditions 9-13 are times following the opposite temperature shift - from 37 to 25°C. The 14 pairwise comparisons in (c) are as follows: 1-7 compare neighbouring time points after an increase in temperature, 8-11 compare neighbouring time points in the decrease in temperature and 12-14 compare the opposing shifts at 15, 30 and 60 mins.

\[
p(a_n; \Theta) = \sum_{k=1}^{K} \pi_k \mathcal{W}(a_n; \mu_k, \beta_k)
\]  

(5.3)

The cluster label for the \(n\)th gene, \(L_n\), is calculated from the angular data, \(a_n\) as follows:

\[
L_n = \arg\max_k \frac{\pi_k \mathcal{W}(a_n; \mu_k, \beta_k)}{p(a_n)}
\]

(5.4)

The next section reviews the theory of circular distributions and how the Wrapped Normal is similar to the Normal (Gaussian) distribution but wrapped around the circumference of a unit circle.

**Circular Distributions**

After the data has been transformed into a series of angles circular distributions are required to appropriately model the underlying components. Two of the most commonly used circular Normal distributions are the von Mises and the wrapped Normal. These two distributions show different desirable statistical properties (Mardia, 1972) but it has been shown that they are close approximations of each other (Stephens, 1963). Therefore, either can be selected and it is common to choose the distribution which is better suited to a particular problem (Mardia, 1972). As calculation of the trigonometric moments is simpler in the the wrapped Normal, this distribution is chosen here.
5.2 Methodology: Data Projections

Wrapped distributions The theory of wrapped distributions is well explained in Fisher (1993); Jammalamadaka and SenGupta (2001); Batschelet (1981) and Mardia (1972). If $X$ is a continuous random variable on the real line with probability density function $g(x)$, then a circular random variable, $A$, is defined by $A = X \mod 2\pi$. The corresponding density function is obtained by wrapping the density $g(x)$ around the circumference of a unit circle and has the following general form:

$$f(a) = \sum_{b=-\infty}^{\infty} g(a + 2\pi b)$$ (5.5)

Therefore, wrapping a univariate Normal distribution around the unit circle gives the density function in equation 5.6, with a mean $\mu'$ and precision $\beta'$.

$$\mathcal{W}\mathcal{N}(a; \mu', \beta') = \frac{\sqrt{\beta'}}{\sqrt{2\pi}} \sum_{b=-\infty}^{\infty} \exp \left( \frac{-\beta'}{2}(a - \mu' - 2\pi b)^2 \right)$$ (5.6)

However, if the variance is relatively small, a large proportion of the of the density value at any value $a$ will be from when $b$ is such that there is the smallest wrapped angular distance between $a$ and $\mu'$. Therefore the infinite sum in equation 5.6 can be approximated by a single choice of $b$, giving equation 5.7 for the univariate case.

$$\mathcal{W}\mathcal{N}(a; \mu', \beta') \approx \frac{\sqrt{\beta'}}{\sqrt{2\pi}} \exp \left( \frac{-\beta'}{2} dw(a, \mu')^2 \right)$$ (5.7)

where $dw(\cdot)$ gives the minimum wrapped distance between two angles such that:

$$dw(a_1, a_2) = \min(|a_1 - a_2|, |a_1 - a_2 + 2\pi|, |a_1 - a_2 - 2\pi|)$$ (5.8)

Only three comparisons are required in equation 5.8 rather than an infinite number as both the data and the mean values in our analysis will be in the range $[0, 2\pi]$. For multivariate data, $dw(\cdot)$ is defined to give a minimum distance vector containing the minimum distances for each dimension. The $D$-variate wrapped Normal density function is approximated by:

$$\mathcal{W}\mathcal{N}(a; \mu', \beta') \approx \frac{|\beta'|^{1/2}}{(2\pi)^{D/2}} \exp \left( -\frac{1}{2} dw(a, \mu')^T \beta dw(a, \mu') \right)$$ (5.9)
Mixtures of Wrapped Normal Distributions

A mixture of $K$ multi-dimensional wrapped Normals is used to model some angular data $A$ such that $A = \{a_{ij}\}$, with $a_{ij} \in [0, 2\pi]$ for $i = 1...N$ and $j = 1..C$, where $C$ is the dimensionality of the angular data (number of pairwise comparisons). The $k^{th}$ wrapped normal component has an associated mixing probability $\pi_k$, mean $\mu_k$ and precision $\beta_k$. $L_n$ is the mixture label of the $n^{th}$ datum.

The model parameters can be inferred from the data using either the Expectation-Maximisation or variational Bayes algorithms. As variational Bayes is more efficient in its choice of model order, this is used. However, an explanation of how to adapt the EM approach and comparison of the EM and VB approaches for circular data is provided in appendix A.

A circular variational Bayes model is similar to a variational Bayes mixture model of Normal distributions, so the same procedure can be used to infer the model. The equations need to be modified slightly to make sure the distances between angles are correctly wrapped. Thus the update equations from Penny (2001) can be modified by the use of variable $z_{nk}^{\mu_k}$ defined in equation 5.10, that gives the wrapped version of angular data for the $n^{th}$ gene closest to $\mu_k$. The modified update equations and model posteriors are also given in appendix A. Figure 5.2 shows an example of circular variational Bayes applied to some toy data containing 3 clusters. The data is wrapped into the range $[0,2\pi]$ along each dimension and circular variational Bayes is used to cluster the data. It correctly predicts a 3 component model.

\[
\begin{align*}
    z_{nk}^{\mu_k} &= \begin{cases} 
        a_{ni} & \text{if } dw(a_n, \mu_k)(i) = |a_{ni} - \mu_k(i)| \\
        a_{ni} + 2\pi & \text{if } dw(a_n, \mu_k)(i) = |a_{ni} + 2\pi - \mu_k(i)| \\
        a_{ni} - 2\pi & \text{if } dw(a_n, \mu_k)(i) = |a_{ni} - 2\pi - \mu_k(i)| 
    \end{cases} 
\end{align*}
\] (5.10)

Initialisation  The mixture model approaches require initialisation of the hyperparameters (or model parameters for circular EM). Given a good initialisation of the parameters they fit a probabilistic model to the distribution of the data by adjusting the parameters until the free energy (or likelihood for EM) has been maximised. However, for high dimensional space, the models can give poor results if initialised badly, for instance with cluster
5.2 Methodology: Data Projections

Figure 5.2: Circular variational Bayes on toy data. (a) data containing 3 clusters, data points plotted as their wrapped version closest to the cluster mean (b) version of data in range \([0, 2\pi]\) (c) posterior mixing probabilities, showing only 3 components predicted in the model (d) expected values and samples taken from the posteriors of the mean and covariance matrices for the 3 components

centres far away from any data points. Thus, if random initialisation is used, multiple initialisations would be required and the model with the maximum negative free energy selected as the global optimal. However, a much better initialisation strategy is to use a method that provides some knowledge of the location of data points in the data space and the relative data densities. Hierarchical/agglomerative clustering provides this information as it builds clusters by initialising with \(N\) groups, centred on the data points, and then the groups are merged according to the distances between them. Thus if a large number of points are close together in the data space these will form a large group in the agglomerative clustering, and data points far away from other data points will form small groups.

By joining clusters until \(K\) clusters remain, and using these groups to get initial estimates for the means, covariances and mixing probabilities, the results are much more stable and only one initialisation and model is required. Wrapped hierarchical clustering was used to initialise the model parameters. Further details can be found in appendix A.
5.3 Methodology: Interpreting the Results

5.2.3 Noise Model Implications for the Microarray Data

Using this mixture of circular distributions to model the data assumes that the noise in the data follows a wrapped Normal distribution in the angular space. This makes the implicit assumption that log fold changes with smaller magnitudes will be generated with smaller absolute values of noise. This is because a smaller amount of noise is required to change the angular value for genes with smaller magnitudes (radial distances) of expression change. Therefore, it should be noted that the noise model in this approach differs from using a GMM approach to directly model the log fold change values.

5.3 Methodology: Interpreting the Results

5.3.1 Locating Interesting Clusters

The pairwise comparisons that are used in the model can be chosen to reflect the purpose of the microarray analysis with each condition in the data set being compared with zero to $D - 1$ other conditions. Making these pairwise comparisons not only allows clusters to be formed from groups of genes showing the same general features of relative expression change, the result of the clustering is much easier to interpret using the associated mean and precision. It is straightforward to label each cluster with a biological interpretation by evaluating its posterior with reference to a set of angles, such as those shown in table 5.1. The precision parameter allows us to determine which ‘features’ (pairwise comparisons) are most informative for a cluster. For example, imagine $a^k_i$ is the angular data for $i$th comparison of genes allocated to cluster $k$, then if the variation of $a^k_i$ is small the genes in cluster $k$ are similar with respect to $i$th comparison. However, a large variation in $a^k_i$ suggests a large spread in gene expression in the experimental conditions. A large variation can occur for two reasons. The first is that genes do not share a conclusive pattern of behaviour with respect to the $i$th comparison, and hence comparison $i$ is not informative for describing the co-regulation of genes in cluster $k$. The second is that the genes show no conclusive change in expression in either conditions used for the $i$th comparison, causing the data to be spread close to the origin, with small radial distances and large variation.
5.3 Methodology: Interpreting the Results

<table>
<thead>
<tr>
<th>Angle</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0000</td>
<td>Up-regulation in condition 1. No change in condition 2.</td>
</tr>
<tr>
<td>0.3927</td>
<td>More up-regulation in condition 1 than condition 2.</td>
</tr>
<tr>
<td>0.7854</td>
<td>Similar up-regulation in condition 1 and condition 2.</td>
</tr>
<tr>
<td>1.1781</td>
<td>More up-regulation in condition 2 than condition 1.</td>
</tr>
<tr>
<td>1.5708</td>
<td>No change in condition 1. Up-regulation in condition 2.</td>
</tr>
<tr>
<td>2.7489</td>
<td>Down-regulation in condition 1. Relatively small up-regulation in condition 2.</td>
</tr>
<tr>
<td>3.1416</td>
<td>Down-regulation in condition 1. No change in condition 2.</td>
</tr>
<tr>
<td>3.5343</td>
<td>More down-regulation in condition 1 than condition 2.</td>
</tr>
<tr>
<td>3.9270</td>
<td>Similar down-regulation in condition 1 and condition 2.</td>
</tr>
<tr>
<td>4.3197</td>
<td>More down-regulation in condition 2 than condition 1.</td>
</tr>
<tr>
<td>4.7124</td>
<td>No change in condition 1. Down regulation in condition 2.</td>
</tr>
<tr>
<td>5.1051</td>
<td>Down-regulation in condition 2. Relatively small up-regulation in condition 1.</td>
</tr>
</tbody>
</table>

Table 5.1: Reference Angles. Suggested interpretations for a set of 16 angles, used to provide a basic summary for each of the pairwise comparisons of a cluster.

in angle. In this case, $r_k^i$ will be have a mean close to zero and an interpretation of no regulatory change either conditions can be made. Providing an automatic description of each cluster that shows under which conditions genes are co-regulated allows the clusters of most relevance to an investigation to be more easily located.

5.3.2 Analysis of Genes Within a Cluster

For each cluster $k$, the $n$th gene has a posterior probability, which gives a measure of how the gene is associated with $k$. For some investigations, such as the heat shock response, the pattern of relative changes in expression is similar but the genes show much variation in the magnitude of expression change. Thus, if the aim of the analysis is to identify genes that may be involved in heat shock then this can be done by locating a cluster corresponding to up-regulation following increase in temperature and then identifying those with the highest probability of being associated with that cluster. However, for some investigations both the pattern of interest and magnitude of intensity change may be of interest. The radial distance, $\hat{r}_n$, of a gene’s expression across the conditions of study can be used as a measure of the magnitude of the expression changes.
Figure 5.3: Probability-radial distance plot. The two measures can be used to visualise the spread of data within a cluster, in this simulated example there are two clusters of genes with associated with the same pattern but different magnitudes of intensity change (radial distance).

\[
\hat{r}_n = \sqrt{\frac{D}{\sum_{t=1}^{n} x_{nt}^2}}
\]  

(5.11)

We also define \(\tilde{r}_n\) as the normalised value of \(\hat{r}_n\), calculated by dividing by the maximum value of \(\hat{r}_n\) for all points allocated to cluster \(k\).

Plots of the radial distance and the posterior probability can be used to investigate sub-clusters within the data. In some data sets there may be groups of genes which share the same pattern but are quite distinct in their magnitude of expression change. If this is the case, the situation can be easily visualised using the two measures, where they will have high probability but form sub groups in the radial distance (figure 5.3). One of the key advantages of this approach is that measures of angular and radial dispersion allow more distinct information relating to gene groups to be revealed.

5.4 Application to Gene Expression Data sets

The method was applied to two microarray data sets, notably the yeast *Saccharomyces cerevisiae* responding to environmental changes (Gasch et al., 2000) and that of germling development in the rice blast fungus *Magnaporthe grisea*.

5.4.1 Yeast Stress Response Data Set

The data set from Gasch et al. (2000) considers the response of 6152 *Saccharomyces cerevisiae* genes to various environmental stresses. Together with work by Causton et al.
5.4 Application to Gene Expression Data sets

(2001), this research highlights a set of global environmental stress response (ESR) genes. In particular, Gasch et al. (2000) highlight approximately 900 ESR genes (283 induced and 583 repressed during stress) following hierarchical clustering analysis, which was subsequently sorted into ‘interesting’ groups by informed visual inspection. The previously identified groups of ESR genes, which have similar regulatory patterns in response to various treatments but are quite varied in their level of intensity change, are used to show that this transformation provides a useful metric by which these sorts of groups can be distinguished.

Interpreting Clusters

Circular Variational Bayes Model of Angular Data To demonstrate the value of this approach 25 conditions were concentrated on, consisting of 8 time points following heat shock from 25 to 37°C, 10 time points following hydrogen peroxide stress and 7 time points following osmotic stress. Genes showing less than a 3-fold change in expression data in all conditions were removed prior to clustering, leaving 2328 genes. All possible pairwise comparisons of neighbouring times points within each time course, heat shock versus hydrogen peroxide treatment and heat shock versus osmotic stress, were used, to examine the relative changes in the expression both between different time points and between different environmental changes.

A circular VB model, initialised with 30 clusters, found 12 clusters in the data. Each cluster was automatically interpreted as a series of pairwise comparisons by observing each element in the mean vector with reference to the set of angles in table 5.1. By observing the variation in the angular data for the genes allocated to a cluster, it was also seen which experimental pairings gave most information regarding the genes within each cluster. For example, table 5.2 shows the reference angles associated with the pairwise comparisons for cluster A and the interpretation of the comparisons. Figure 5.4 shows the variation of angular data for clusters A and B. Cluster A shows small variation in the majority of the pairwise comparisons and combining this with the cluster interpretation we can conclude that cluster A contains genes induced following all three types of shock. Cluster B,
### Table 5.2: Example of interpretation for the pairwise comparisons for cluster A using the reference angles in table 5.1. HS = heat shock from 25°C to 37°C, HP = constant 0.32 mM H2O2 treatment, OS = Osmotic stress treatment (1M sorbitol)

<table>
<thead>
<tr>
<th></th>
<th>Expected value of mean</th>
<th>Most probable reference angle</th>
<th>Interpretation of angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.95706</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 5 minutes and HS 10 minutes</td>
</tr>
<tr>
<td>2</td>
<td>0.98396</td>
<td>1.1781</td>
<td>More up-regulation in HS 15 minutes than HS 10 minutes</td>
</tr>
<tr>
<td>3</td>
<td>0.73564</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 15 minutes and HS 20 minutes</td>
</tr>
<tr>
<td>4</td>
<td>0.71957</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 20 minutes and HS 30 minutes</td>
</tr>
<tr>
<td>5</td>
<td>0.47394</td>
<td>0.3927</td>
<td>More up-regulation in HS 30 minutes than HS 40 minutes</td>
</tr>
<tr>
<td>6</td>
<td>0.55832</td>
<td>0.3927</td>
<td>More up-regulation in HS 40 minutes than HS 60 minutes</td>
</tr>
<tr>
<td>7</td>
<td>0.56407</td>
<td>0.3927</td>
<td>More up-regulation in HS 60 minutes than HS 80 minutes</td>
</tr>
<tr>
<td>8</td>
<td>1.0143</td>
<td>1.1781</td>
<td>More up-regulation in HP 20 min than HP 10 min</td>
</tr>
<tr>
<td>9</td>
<td>0.92699</td>
<td>0.7854</td>
<td>Similar up-regulation in HP 20 min and HP 30 min</td>
</tr>
<tr>
<td>10</td>
<td>0.49963</td>
<td>0.3927</td>
<td>More up-regulation in HP 30 min than HP 40 min</td>
</tr>
<tr>
<td>11</td>
<td>1.0106</td>
<td>1.1781</td>
<td>More up-regulation in HP 50 min than HP 40 min</td>
</tr>
<tr>
<td>12</td>
<td>0.64163</td>
<td>0.7854</td>
<td>Similar up-regulation in HP 50 min and HP 60 min</td>
</tr>
<tr>
<td>13</td>
<td>0.55943</td>
<td>0.3927</td>
<td>More up-regulation in HP 60 min than HP 80 min</td>
</tr>
<tr>
<td>14</td>
<td>0.49968</td>
<td>0.3927</td>
<td>More up-regulation in HP 80 min than HP 100 min</td>
</tr>
<tr>
<td>15</td>
<td>0.87785</td>
<td>0.7854</td>
<td>Similar up-regulation in HP 100 min and HP 120 min</td>
</tr>
<tr>
<td>16</td>
<td>0.63378</td>
<td>0.7854</td>
<td>Similar up-regulation in HP 120 min and HP 160 min</td>
</tr>
<tr>
<td>17</td>
<td>1.1838</td>
<td>1.1781</td>
<td>More up-regulation in OS 15 min than OS 5 min</td>
</tr>
<tr>
<td>18</td>
<td>0.76818</td>
<td>0.7854</td>
<td>Similar up-regulation in OS 15 min and OS 30 min</td>
</tr>
<tr>
<td>19</td>
<td>0.55894</td>
<td>0.3927</td>
<td>More up-regulation in OS 30 min than OS 45 min</td>
</tr>
<tr>
<td>20</td>
<td>0.049467</td>
<td>0.000</td>
<td>Up-regulation in OS 45 min. No change in OS 60 min</td>
</tr>
<tr>
<td>21</td>
<td>1.3835</td>
<td>1.5708</td>
<td>No change in OS 60 min. Up-regulation in OS 90 min</td>
</tr>
<tr>
<td>22</td>
<td>1.1598</td>
<td>1.1781</td>
<td>More up-regulation in OS 120 min than OS 90 min</td>
</tr>
<tr>
<td>23</td>
<td>0.48553</td>
<td>0.3927</td>
<td>More up-regulation in HS 10 minutes than HP 10 min</td>
</tr>
<tr>
<td>24</td>
<td>0.55983</td>
<td>0.3927</td>
<td>More up-regulation in HS 20 minutes than HP 20 min</td>
</tr>
<tr>
<td>25</td>
<td>0.66192</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 30 minutes and HP 30 min</td>
</tr>
<tr>
<td>26</td>
<td>0.59304</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 40 minutes and HP 40 min</td>
</tr>
<tr>
<td>27</td>
<td>0.9212</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 60 minutes and HP 60 min</td>
</tr>
<tr>
<td>28</td>
<td>0.82436</td>
<td>0.7854</td>
<td>Similar up-regulation in HS 80 minutes and HP 80 min</td>
</tr>
<tr>
<td>29</td>
<td>0.4625</td>
<td>0.3927</td>
<td>More up-regulation in HS 5 minutes than OS 5 min</td>
</tr>
<tr>
<td>30</td>
<td>0.48755</td>
<td>0.3927</td>
<td>More up-regulation in HS 15 minutes than OS 15 min</td>
</tr>
<tr>
<td>31</td>
<td>0.52756</td>
<td>0.3927</td>
<td>More up-regulation in HS 30 minutes than OS 30 min</td>
</tr>
<tr>
<td>32</td>
<td>0.13621</td>
<td>0.000</td>
<td>Up-regulation in HS 60 minutes. No change in OS 60 min</td>
</tr>
</tbody>
</table>
### Table 5.3: Yeast data, circular VB analysis

The distribution of the ESR genes amongst the clusters highlighted in Gasch et al. (2000) is shown, as is a summary of pairwise comparisons and interpretations relevant to clusters A-F, found by analysis of the expected mean and the variation in the angular data for each cluster. The interpretations highlight the differences between clusters in terms of relative changes between conditions. (HS = heat shock, HP=hydrogen peroxide, OS = osmotic stress, RMD=set of genes removed prior to clustering)

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Size</th>
<th>Common features (pairwise comparisons)</th>
<th>Interpretation of cluster</th>
<th>Induced ESR</th>
<th>Repressed ESR</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>504</td>
<td>All apart from OS 60 vs 90 mins</td>
<td>Induction in HS, HP &amp; OS</td>
<td>248 (87.6%)</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>372</td>
<td>Only HS comparisons</td>
<td>Induction in HS</td>
<td>12 (4.0%)</td>
<td>4 (0.7%)</td>
</tr>
<tr>
<td>C</td>
<td>73</td>
<td>All apart from HP 40-60 mins and OS &gt;45mins</td>
<td>Induction in HS, HP &amp; OS, repression in HP&gt;60min</td>
<td>7 (2.5%)</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>486</td>
<td>Only HS comparisons</td>
<td>Repression in HS</td>
<td>1 (0.4%)</td>
<td>33 (5.6%)</td>
</tr>
<tr>
<td>E</td>
<td>695</td>
<td>All apart from those of OS &gt;45 mins</td>
<td>Repression in HS, HP &amp; OS</td>
<td>445 (76%)</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>139</td>
<td>All apart from comparisons at times &gt;40 mins</td>
<td>Repression during intial 40 mins in HS, HP &amp; OS</td>
<td>39 (6.7%)</td>
<td></td>
</tr>
<tr>
<td>G-L</td>
<td>59</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMD</td>
<td>3824</td>
<td></td>
<td></td>
<td>15 (5.6%)</td>
<td>64 (11%)</td>
</tr>
</tbody>
</table>
Figure 5.4: Yeast data: box plots of the angular data for genes allocated to clusters A and B across the 32 pairwise comparisons. Comparisons 1-7 correspond to heat shock (HS), 8-16 hydrogen peroxide and 17-22 osmotic stress (OS) time-points, 23-28 heat shock versus hydrogen peroxide and 29-32 heat shock versus hydrogen peroxide. Cluster A contains genes whose angular data is tightly clustered in most pairwise comparisons. This cluster shows a pattern of induction (with angles between 0-1.58) in all three types of shock, although the genes are not so well grouped in the comparison of 60 and 90 mins after osmotic stress (comparison 21). Cluster B is more tightly grouped in the comparisons of heat shock (HS, 1-7) than the other types of shock, corresponding to a pattern of induction only following heat shock.

however, reveals large variation in all comparisons other than those of heat shock (1-7) and as such indicates a pattern of induction only following heat shock. Therefore, if the purpose of the analysis was to study global response genes, then cluster A would merit further investigation whereas if genes induced only following heat shock where under investigation then cluster B would be of most interest. Table 5.3 provides a summary of conclusions made in the same manner for clusters A-F. Table 5.3 also summarises the distribution of the previously identified ESR genes amongst these clusters, the majority of which are found in the two large clusters (A and E) of induced and repressed genes, conforming to the findings of Gasch et al. (2000). The small proportion (4.0%) of the ESR genes in cluster B are likely to have been misclassified as showing a global ESR response in the previous study (Gasch et al., 2000) as this probabilistic model allocates them with much greater probability to a heat shock only response.

**Variational Bayes With Standardised Data**  A VB mixture model was applied to the standardised log fold change data for the same 2328 genes used in the circular VB model across the 25 conditions of heat shock, hydrogen peroxide and osmotic stress. In contrast to the circular VB model which distinguished only 12 clusters, genes were allocated to all 30 components that the model was initialised with, suggesting less structure in the stan-
5.4 Application to Gene Expression Data sets

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Size</th>
<th>Induced ESR</th>
<th>Repressed ESR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>571</td>
<td>237 (83.8%)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>143</td>
<td>3 (1.1%)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>189</td>
<td>19 (6.7%)</td>
<td>1 (0.2%)</td>
</tr>
<tr>
<td>5</td>
<td>60</td>
<td>1 (0.4%)</td>
<td>5 (0.9%)</td>
</tr>
<tr>
<td>6</td>
<td>104</td>
<td>7 (2.5%)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>6</td>
<td>1 (0.2%)</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>269</td>
<td>16 (2.7%)</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>22</td>
<td>1 (0.2%)</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>849</td>
<td>492 (84.4%)</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>1 (0.2%)</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>36</td>
<td>1 (0.4%)</td>
<td>3 (0.5%)</td>
</tr>
<tr>
<td>28</td>
<td>30</td>
<td>1 (0.2%)</td>
<td></td>
</tr>
<tr>
<td>RMD</td>
<td>3824</td>
<td>15 (5.6%)</td>
<td>64 (11%)</td>
</tr>
</tbody>
</table>

Table 5.4: Yeast data, VB analysis of standardised data: The distribution of the ESR genes amongst the clusters highlighted in Gasch et al. (2000) is shown. Genes were allocated to all 30 clusters in the model, only 12 contained ESR genes, the other 18 contained a total of 29 genes. (RMD = set of genes removed prior to clustering)

Standardised data. However, analysis of the ESR genes in this model did largely conform to the previous findings of Gasch et al. (2000) with the majority of the induced genes clustered together and the majority of the repressed genes clustered together (table 5.4). This suggests that the standardised data does contain some useful information with which to locate these interesting regulatory changes. A major disadvantage of standardisation is that it transforms data to a representation in which similar ‘shaped’ expression profiles are grouped together but - by contrast to the angular transform - it is difficult to interpret the underlying ‘shape’ of each cluster. Furthermore, it is difficult to identify what differences in regulation distinguish different clusters. For example, it is difficult to determine why 19 of the induced ESR genes were considered more likely to be in cluster 4 rather than in cluster 1 with the majority of the induced ESR genes (figure 5.5). Finding interesting clusters and target genes is therefore more difficult.

Within Cluster Analysis

Figure 5.6 shows how the genes in cluster A are spread in terms of their probability and radial distance. It can be seen that all genes in this cluster have high probability (with a mean of 0.998) and therefore show a pattern of induction in the environmental stress
5.4 Application to Gene Expression Data sets

Figure 5.5: Yeast data: two clusters found with a VB model of standardised data. 239 of the induced ESR genes are in cluster 1 and 19 are found in cluster 4. Standardisation groups genes with similar shaped expression profiles but it is difficult to interpret the 'features' of the shape that define a cluster. Plots of the standardised data, box plots showing the spread of the standardised data and plots of the log fold changes are shown for the two clusters but none provide information as to the differences in regulation that distinguish the two clusters.

Figure 5.6: Yeast data: probability-radial distance plots for all the genes in cluster A and only the ESR genes in cluster A. Both plots show that the genes are tightly grouped in their posterior probability but are distributed amongst a wide range of intensity changes. The ESR genes previously categorised are well spread throughout the range of radial distances, so these response genes are similar in their shape but not intensity of expression change.

5.4.2 Rice Blast Data Set

This data set followed changes in transcript abundance levels of 13,666 expressed sequence tags (ESTs) from the fungus *Magnaporthe grisea* during infection-related development.
Table 5.5: Rice blast data: the four pairwise comparisons chosen to mine the data for putative pathogenicity factors. (I= inductive surface, NI=non inductive surface, 7h = 7 hours post inoculation, 12h = 12 hours post inoculation, spore = ungerminated spore sample at time 0)

The five conditions of study in this microarray data set represent the ungerminated spore sample, at 7 hours and 12 hours post inoculation on a hydrophobic surface and the same time points post inoculation on a hydrophilic surface as detailed in Dean et al. (2005) (figure 2.14). Lowess regression was applied to each array to remove dye bias (Yang et al., 2002) (further details in appendix B) and the log ratio data calculated. 2968 genes were then selected for further study as those that showed a 2-fold or larger expression change in at least one condition. Positive log ratio values imply the gene has a higher expression in the condition after germination than the spore sample (up-regulation) and negative values imply there is higher expression in the spore sample (down-regulation).

In the analysis of this data set, it is of interest to find genes differentially expressed between time points and surface types. Thus, the expression ratio data for the selected 2968 genes was transformed into 4-dimensional angular data by making the pairwise comparisons shown in table 5.5. In the absence of experimental repeats, which are very expensive to obtain, co-regulated groups of genes can be located and those groups showing larger expression in the hydrophobic conditions selected for further study.

**Results**

A Variational Bayes model, initialised with 30 clusters, revealed a 9-component model to describe this data set. Two of the clusters (cluster A and B) contained genes who expression was highest during the conditions of appressorium formation and plant penetration. Amongst this cohort will be genes pivotal to the disease process, termed pathogenicity factors.
Cluster A contains the genes up-regulated against spore for both time points on the inductive surface, with greater up-regulation at 12 hours than 7 hours but relatively little change in regulation for the non-inductive surface time points (figure 5.7, table 5.6). By observing box plots of the angular data for each pairwise comparison we can see that comparison two has large variation but the data for the other three comparisons is quite tightly grouped. The variation in the second comparison occurs because there are only small regulatory changes in the hydrophilic conditions, so the data is close to the centre with much more variation in angle. The mean for this pairwise comparison is therefore not useful in defining the expression pattern corresponding to the cluster.

Cluster B shows the least variation in angles 1 and 4 (figure 5.7), showing that the genes were tightly clustered in the comparisons against 12h I. The interpretation of the cluster corresponds to large up-regulation in 12h I, the condition associated with rice plant infection, with no regulatory change in 7h I and relatively small down-regulation in 12h NI (table 5.6).

Within clusters A and B, there are genes that have already drawn attention as prime candidates for gene-knockout studies in the quest to understand the infection process of rice pathogen (Dean et al., 2005). For instance, cluster A contained two putative cutinases MG09100 and MG02393. Cluster B contained MG10072, a putative polyketide synthase, cutinase MG11108 and a CFEM-like gene MG09863. As well as these prime candidates with high posterior probability (> 0.9) of association with the clusters many other genes are also contained that have not been previously investigated as putative pathogenicity determinants. In fact, it is notable that the 68% of cluster A and 71% of cluster B have unknown Pfam annotations.

Cluster C, interpreted as similar up-regulation in each condition against the spore sample (table 5.6), is the largest cluster, containing 1150 genes. Found within this cluster are two polyketide synthases, MG07775 and MG07219, two CFEM-like genes MG05871 and MG09022 and the cutinase MG02301, which were also highlighted in Dean et al. (2005). Although the results suggest that these genes do not show a significantly different expres-
### Cluster A

<table>
<thead>
<tr>
<th>Mean</th>
<th>Ref. Ang</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 1.0326</td>
<td>1.1781</td>
<td>More up-regulation in I 12h/sp than I 7h/sp</td>
</tr>
<tr>
<td>2 2.7802</td>
<td>2.7489</td>
<td></td>
</tr>
<tr>
<td>3 6.1439</td>
<td>0.000</td>
<td>Up-regulation in I 7h/sp. No change in NI 7h/sp</td>
</tr>
<tr>
<td>4 0.038273</td>
<td>0.000</td>
<td>Up-regulation in I 12h/sp. No change in NI 12h/sp</td>
</tr>
</tbody>
</table>

### Cluster B

<table>
<thead>
<tr>
<th>Mean</th>
<th>Ref. Ang</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 1.6403</td>
<td>1.5708</td>
<td>No change in I 7h/sp. Up-regulation in I 12h/sp</td>
</tr>
<tr>
<td>2 3.4124</td>
<td>3.5343</td>
<td></td>
</tr>
<tr>
<td>3 4.4131</td>
<td>4.3197</td>
<td></td>
</tr>
<tr>
<td>4 6.0841</td>
<td>5.8905</td>
<td>Up-regulation in I 12h/sp, Relatively small down-regulation in NI 12h/sp</td>
</tr>
</tbody>
</table>

### Cluster C

<table>
<thead>
<tr>
<th>Mean</th>
<th>Ref. Ang</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0.7086</td>
<td>0.7854</td>
<td>Similar up-regulation in I 7h/sp and I 12h/sp</td>
</tr>
<tr>
<td>2 0.70113</td>
<td>0.7854</td>
<td>Similar up-regulation in NI 7h/sp and NI 12h/sp</td>
</tr>
<tr>
<td>3 0.90268</td>
<td>0.7854</td>
<td>Similar up-regulation in I 7h/sp and NI 7h/sp</td>
</tr>
<tr>
<td>4 0.92429</td>
<td>0.7854</td>
<td>Similar up-regulation in I 12h/sp and NI 12h/sp</td>
</tr>
</tbody>
</table>

### Cluster D

<table>
<thead>
<tr>
<th>Mean</th>
<th>Ref. Ang</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 3.9527</td>
<td>3.927</td>
<td>Similar down-regulation in I 7h/sp and I 12h/sp</td>
</tr>
<tr>
<td>2 3.86</td>
<td>3.927</td>
<td>Similar down-regulation in NI 7h/sp and NI 12h/sp</td>
</tr>
<tr>
<td>3 3.9343</td>
<td>3.927</td>
<td>Similar down-regulation in I 7h/sp and NI 7h/sp</td>
</tr>
<tr>
<td>4 3.8477</td>
<td>3.927</td>
<td>Similar down-regulation in I 12h/sp and NI 12h/sp</td>
</tr>
</tbody>
</table>

Table 5.6: Rice blast data: The interpretation for clusters A-D. The interpretation is not shown for the pairwise comparisons that have relatively large variations in their angular data (figure 5.7).

At 12 hours on the inductive surface, they are all notably induced during germling development and may therefore still be necessary for virulence attributes. By contrast, Cluster D, containing 1027 genes, is interpreted as showing similar down-regulation in all four conditions from the spore sample (5.6). The large number of genes in clusters C and D highlight the major changes in gene expression required during germling morphogenesis.

**Experimental Investigation of Cutinases in Clusters A and B** The regulation, during development on a hydrophobic surface, of the three cutinase encoding genes found within clusters A and B have been investigated with more thoroughly with more robust experimental techniques (Skamnioti and Gurr, 2007) over a wider selection of time points.
Figure 5.7: Rice blast data for clusters A-D: log fold data for the genes in each cluster across the four pairwise comparisons and boxplots showing spread of angular data in each cluster. Cluster A shows large variation in the angular data for the second pairwise comparison, suggesting that no pattern of regulatory change in either NI 12h or NI 7h for the genes in this cluster. Cluster B shows the least variation in angles 1 and 4 which corresponds to a pattern of up-regulation in 12h I, with no change in 7h I and relatively small down-regulation in NI 12h. Cluster C shows data centred around 0.71-0.92 for the pairwise comparisons, corresponding to similar up-regulation against spore for all four conditions (see tables 5.1). Cluster D shows very small variation in the data, grouped close to 3.9 radians for all four pairwise comparisons which corresponds to similar down-regulation in all conditions against the spore sample.
5.4 Application to Gene Expression Data sets

Figure 5.8: Transcript analysis of three cutinase encoding genes at various time points on a hydrophobic surface (MG02393, MG09100 and MG11108 were renamed MGG02393, MGG09100 and MGG11966 respectively in genome release 5). Quantitative real-time PCR was used to monitor the transcript abundance (TA) relative to two constitutively expressed genes, ECG1 and βTUB. MGG02393 and MGG09100 showed great up-regulation at 12 hours with some up-regulation also occurring at previous time points (agreeing with the expression pattern associated with cluster A). Whereas, MGG11966 shows only a negligible increase in expression until 12 hours, agreeing with the expression pattern associated with cluster B. Figure reproduced from Skamnioti and Gurr (2007).

MG09100 and MG02393 were found in cluster A, and thus associated with up-regulation compared to the spore for both for both time points on the inductive surface, but with greater up-regulation at 12 hours than 7 hours. Meanwhile, MG11108 was in cluster B, and was thus associated with an expression pattern of large up-regulation at 12 hours but no regulatory change at 7 hours compared with the spore sample. The results of the transcript analysis study by Skamnioti and Gurr (2007) confirmed these patterns of expression (figure 5.8). Furthermore, the gene knockout experiment of MG09100 (termed cut2 in Skamnioti and Gurr (2007)) caused the appressorium not to develop even on the hydrophobic plant surface, and thus is likely to play an important role in the complex signalling between plant surface and fungus (figure 5.9).
5.5 Summary

The example of yeast heat shock treatments have shown that the intensity changes of gene expression profiles are not necessarily the metric of interest when trying to locate groups of co-regulated genes. Instead, this chapter has presented a method that selects the interesting ‘features’ of expression patterns shared by co-regulated genes as their relative changes in expression between differing conditions. The method works by performing a series of two data projections. Firstly, pairwise comparisons of expression in different conditions are transformed into an angular representation and then a mixture model of circular distributions is used to cluster the genes according to their transformed data. This methodology can be alternatively thought of as decomposing the data into angular structures.

The method has been compared with other clustering techniques using a set of various environmental stresses of yeast. The variational Bayes approach is shown to be superior to the hierarchical clustering analysis, allowing sub-groups to be automatically distinguished in the context of a fully probabilistic model that also provides a computationally efficient model order selection. The angular transformation of gene expression data is superior to other transformations, such as standardisation because: (1) the clustering is done on specific ‘features’ of the expression rather than the overall shape (2) the choice of ‘features’ can
be selected in an intelligent manner to reflect the questions that the analysis is attempting to address, allowing control over the definition of similarity between genes (3) the angular transformation provides a metric from which some useful information about relative regulation changes, can be automatically mined and used to understand the ‘features’ important to different clusters.

The stress response in yeast has been well studied in previous work (Gasch et al., 2000; Causton et al., 2001; Gasch and Werner-Washburne, 2002) but little is currently known about the interaction of the rice blast fungus with its host during pathogenic development. The technique was applied to a dataset of Magnaporthe grisea in order to identify potential pathogenicity factors. Indeed, two of the clusters found could be of particular importance in the study of rice plant infection and may be used to direct the biologist towards specific target genes, which may play pivotal roles in pathogenicity. Some of the small number of genes already associated with pathogenic behaviour of this fungus (Dean et al., 2005) were located at the centre of the clusters corresponding to up-regulation on an inductive surface, and gene knockout studies of three cutinases have confirmed the expression patterns suggested in the microarray data set and their association with the expression patterns of clusters A and B.
6

Visualisation of Genetic Information on a 2D Map

6.1 Introduction

Chapter 4 investigated the structures present in the rice blast and yeast data sets. Chapter 5 developed a technique to locate these structures and describe them in terms of pairwise comparisons of expression change. This chapter shows the results of visualisation of the data sets using the GTM algorithm to display the gene expression on the idea of a 2D map introduced in chapter 3 and also how the information from the variational Bayes clustering can be used to aid the visualisation. It shows how information such as functional annotations and protein family annotations can be highlighted, enabling the spread of these gene groups across the expression patterns to be seen. Substructure/subgroups can also be visualised by observing the GTM plot for each cluster individually. For the two data sets studied in this thesis the angular analysis seemed to provide interesting gene clusters, however for other data sets it may be more useful to perform variational Bayes clustering on the log ratio expression data if it is of interest to group genes with similar intensities of expression change. As the different analyses provide information about different types of groupings within the data and the method is computationally efficient there is no reason not to perform both sorts of analyses and visualise the differences, so both are demonstrated. This chapter first discusses how to modify the GTM algorithm in order to visualise the angular transformed version of the data, by considering the affect of wrapping the data space and latent space.
6.2 GTM for Angular Data

GTM is effectively a constrained mixture model in which the mean the of Gaussian distributions are linked by a mapping from latent space to data space, where this mapping is in the form of a radial basis function (RBF) network. The modifications made to the Gaussian mixture modelling in chapter 5 in order to model mixtures of wrapped Normal distributions can also be applied to the GTM algorithm such that GTM can be used with angular data.

Wrapping the Data Space

The major modification required is to substitute the Gaussian density function with the wrapped normal density function, calculating the distance between the data points and the mean of the wrapped normal as the minimum wrapped angular distance (see equation 5.9).

Thus, the probability of an angular data point \( a_n \) being associated with the \( i \)th latent variable is given by equation 6.1 for angular GTM (rather than equation 3.4 for non circular data). The definition of \( dw(\cdot) \) is given in equation 5.8, \( W \) is the weight matrix for the RBF network, \( \beta \) is the inverse covariance parameter for the wrapped normal distributions, \( C \) is the dimensionality of the circular data, \( K \) is the number of latent variables and \( N \) is the number of data points in the data set.

\[
p(a_n|g_i, W, \beta) \approx \left( \frac{\beta}{2\pi} \right)^{C/2} \exp \left( -\frac{1}{2} dw(a_n, h(g_i; W)^\prime)^T \beta dw(a_n, h(g_i; W)^\prime) \right)
\] (6.1)

Therefore, the maximum likelihood equation for angular GTM is given by:

\[
L(W, \beta) = \sum_{n=1}^{N} \ln \left\{ \frac{1}{K} \left( \frac{\beta}{2\pi} \right)^{C/2} \sum_{i=1}^{K} \exp \left( -\frac{1}{2} dw(a_n, h(g_i; W)^\prime)^T \beta dw(a_n, h(g_i; W)^\prime) \right) \right\}
\] (6.2)

The update equation for the covariance parameter \( \beta \) should also be modified to use the minimum wrapped distance between data points and the centres of wrapped normal distributions, such that,
6.2 GTM for Angular Data

\[ \beta^{-1} = \frac{1}{NC} \sum_{l=1}^{K} \sum_{n=1}^{N} R_{ln}(W, \beta) dw(h(g_l ; W))^2 \]  

(6.3)

and where \( R_{ln}(W, \beta) = \frac{p(a_n | g_l, W, \beta)}{\sum_{i=1}^{K} p(a_n | x_i, W, \beta)} \), which gives the responsibility for mixture component \( l \) for data point \( n \).

The update calculation for \( W \) also needs to be adapted. For a non-angular data set \( X \), \( W \) is calculated using the relationship

\[ \phi^T Q \phi W^T = \phi^T RX \]

where \( Q \) is a diagonal matrix with elements \( Q_{ll} = \sum_{n=1}^{N} R_{ln}(W, \beta) \). However, for an angular data set, \( A \), the equation \( RX \) needs to substituted with the matrix \((RA)^*\), in which each row is calculated with the appropriately wrapped version of the data. For example, the \( i \)th row of \( RA \) would give an estimate of \( h_i \) as the mean position, according to the positions of the data points and the current responsibilities of the \( i \)th latent variable for the data points, calculated as

\[ (RA)_i = \sum_n R_{in} a_n \]

However, if the angles are not correctly wrapped then the average position for \( h_i \) will not accurately reflect the position in wrapped space. Therefore, the \( i \)th row of \((RA)^*\) needs to be calculated using the version of the angular data closest to the current estimate of \( h_i \).

Defining \( z_{h_i}^n \) to the wrapped version of the \( n \)th data point closest to \( h_i \),

\[ (RA)^*_i = \sum_n R_{in} z_{h_i}^n \]

or in vector form,

\[ (RA)^*_i = R_i Z^{h_i} \]
Wrapping the Latent Space

To wrap the latent space the activations of each radial basis function for points in latent space can be calculated using the wrapped distance, $dw(\cdot)$, defined in equation 5.8. The activation for point $g$ in the $m$th basis functions can therefore be calculated with equation 6.4. Points in the latent space are then taken from the range $[0, 2\pi]$.

\[
\phi_m(g) = \exp\left\{-\frac{dw(g, \mu_m)^2}{2\sigma^2}\right\} \tag{6.4}
\]

Wrapping a 1D latent space in this way effectively means that the GTM tries to fit a circle to the data (like stretching a rubber band around the data). In 2D, this would mean fitting a potentially deformed sphere to the data, or if the latent space was only wrapped in one direction, then fitting a cylinder to the data. If this is the type of data being modelled, then this wrapping of latent space could be potentially useful, however manifolds or clusters of data that appear in angular space do not necessarily have these circles and so this wrapping will incorrectly fit the data. Thus, for our angular GTM analysis we need only wrap the data space.

1D Examples

Figure 6.1 shows 1D GTM manifolds fitted to two simple data sets, one of a curve in wrapped space and one containing four clusters. Without the modifications, GTM does not take into account the wrapped distances between points and incorrectly models data at the opposite edges of the data space with different parts of the manifold (see figure 6.1(b)). With the modification to wrap the data space the manifold does not model the wrapped clusters as two separate clusters (figure 6.1(c)), it may appear that the GTM manifold has not been fitted to all the data, but because the data space is wrapped the points are being modelled with the correct part of the manifold. The angular GTM manifold for the curve data is not the optimal manifold that could have been chosen to fit the data, this is probably due to the initialisation used. The GTM results with wrapped latent spaces (figure 6.1(d) and (e)) do not model the spread of data that we require, as it tries to fit a closed circle to
the data.

6.3 Visualisation of the Microarray Data sets

GTM can be used to provide a 2D plot of the data points either from the reduction of the expression data, $X$; or from the reduction of the angular data, $A$, using the angular GTM algorithm. However, trying to interpret the GTM plots on their own can be difficult, it is not always easy to distinguish the groups and patterns of data points once they have been projected into the 2D space (figures 6.2 and 6.3 show angular and non-angular GTM reductions for the rice blast and yeast data sets). Magnification factors can be used to provide some information about the local stretching of the manifold, in which the least stretching occurs in areas of high data density and vice versa. However, even with the magnification factors it can still be difficult to decipher any underlying groupings (figures 6.2 and 6.3).

In order to make gene groupings clearer, previous visualisation techniques have been used in conjunction with clustering methods. For instance, the heatmaps are often used in conjunction with a hierarchical clustering tree to order the genes into groups and expression profile plots can used with a k-means clustering result, in which the profiles for each cluster are plotted separately. In a similar manner, information from the variational Bayes clustering can be used to aid the visualisation of patterns in the 2D plot provided by the GTM. Other information about the genes, such as annotations and subsets of genes can also be easily highlighted in this sort of visualisation.
Figure 6.1: Comparison of GTM with different modifications on simple data sets. (a) two data sets with wrapped data in the range \([0, 2\pi]\). GTM results on the data sets with (b) no modifications (c) wrapped data space modification (angular GTM) (d) wrapped latent space modification (e) wrapped latent and wrapped data space modifications.
Figure 6.2: Plots and magnification factors from the GTM reductions of the rice blast data set. (a) non-angular, reduced from 4 dimensions and (b) angular analysis reducing the data from 4 dimensions. Some areas of the plots have higher data density, suggesting groupings of data points but it is difficult to extract patterns even with the magnification factors as there is no information about which regions of the plot are related to which types of expression changes. Using the GTM to visualise the information without incorporating other information is therefore of limited value.
Figure 6.3: Plots and magnification factors from the GTM reductions of the yeast data sets. (a) non-angular reducing the dimension from 18 dimensions and (b) angular analysis reducing the data from 32 dimensions. Like the plots for the rice blast data in figure 6.2, some areas of the plots have higher data density, suggesting groupings of data points but it is difficult to extract patterns. It is particularly difficult to see any patterns or structure in the visualisation of the non-angular data set. There seems to be more structure in the visualisation of the angular data, despite being reduced from higher dimensions.
The idea can be defined more formally in terms of information relating to the microarray data set from various projections of the data; information relating to specific genes and information about potentially interesting subsets of genes. Associated with the microarray data set there can be cluster parameters from variational Bayes clustering of the expression matrix, $X$ and from circular variational Bayes clustering of the angular data $A$. The expected values of these parameters for these two models can be denoted as $\Theta^X = \{\pi^X, \mu^X, \beta^X\}$ and $\Theta^A = \{\pi^A, \mu^A, \beta^A\}$ respectively. If GTM is used to project the expression data and angular data into 2D space, then the mappings $h^X(g; W)$ and $h^A(g; W)$ are also found.

Associated with the data set is a set of $N$ genes, which can be denoted by $\Gamma$. For the $n$th gene in $\Gamma$, $\gamma_n$, is therefore log fold data, $x_n$, angular data, $a_n$, cluster labels from variational Bayes analysis $L_n^X$ or circular variational Bayes, $L_n^A$, and a radial distance $r_n$. Each gene also has co-ordinates in 2D space, $Y_n^X$ and $Y_n^A$ from the GTM analysis. Each gene $\gamma_n$, may also have other information such as Protein Family annotations (PFAM), functional annotations, or ontological information, where $\psi_{ni}$ denotes the annotation for gene $\gamma_n$ from the $i$th annotation source. There may also pre-identified subsets of genes of interest that are of interest to highlight. $\Gamma^j$ is defined to be the $j$th subset that is of interest to highlight.

In order to understand the patterns of expression change present in the data set, it is of interest to visualise as much of this information as possible (see figure 6.4) as each source of information informs about a different aspect of the data set or current biological understanding. Thus, this chapter demonstrates how the different types of information can be visualised to provide some insight into the distribution of genes across expression patterns.

### 6.3.1 Visualisation of Patterns Using Cluster Information

The Variational Bayes clustering results provide information about the underlying patterns/groupings in the data as the method fits an unconstrained mixture model to the points. The cluster labels as well the cluster parameters can therefore be used to aid the visualisation of the patterns.
6.3 Visualisation of the Microarray Data sets

Figure 6.4: Different types of information regarding the data set which may be of interest to visualise. (a) Visualisation of the log expression values for each gene, \( x_n \), can be mapped into 2D co-ordinates \( y^X_n \) using the GTM algorithm. Each gene may also have associated cluster label from variational Bayes clustering of \( X \), a radial distance and a vector of annotations \( \psi_n \). To improve the visualisation of the clusters, it may also be useful to visualise information about the expected values of the mixture model parameters \( \mu^X_1, \ldots, \mu^X_K \) and precision matrices \( \beta^X_1, \ldots, \beta^X_K \). It may also be of interest to highlight selected subsets of the genes to observe their distribution through the different clusters. (b) The visualisation of angular transformed data can be done in the same way as the expression data, except that a version of the GTM modified for use with angular data is required to project each angular data point \( a_n \) to 2D co-ordinates \( y^A_n \). The cluster labels and expected value of the mixture model parameters \( \mu^A_1, \ldots, \mu^A_K \) and precision matrices \( \beta^A_1, \ldots, \beta^A_K \) need to be calculated using the circular variational Bayes method described in chapter 5.
Cluster labels and Cluster Means

Figures 6.5 and 6.6 show the same 2D maps as figures 6.2 and 6.3 but the genes are highlighted according to their cluster labels from the variational Bayes analysis. It can been seen how the manifold have been fitted around the groups of genes found by the unconstrained mixture model approach. The mean of each cluster (expected value of the $\mu_k$'s) have been projected in 2D space to show the positions of the cluster centres in relation to the 2D manifold.

The variational Bayes analysis also provides information about the expression patterns associated with each cluster, either from the mean intensities for the non-angular analysis or from the interpretation of the angles in terms of pairwise comparisons for the angular analysis. Thus, using the information from the variational Bayes clustering it is possible to associate different areas of the plot with different patterns of expression change.

Figure 6.5 shows the cluster labels and projected cluster means for the rice blast data set. Both the non-angular and angular analysis show reasonably good separation of the clusters. However, for the yeast data set (figure 6.6), the non-angular analysis did not show any correspondence between the variational Bayes clusters and the area of the manifold used to model the data points, making it difficult to associate any areas of the plot or groups of data points with particular expression changes. The correspondence between the GTM visualisation and the clusters from the angular analysis were much clearer.
Figure 6.5: Visualisation of rice blast data with cluster labels, projected cluster means and projected eigenvectors of the cluster covariance matrices. Different cluster labels are shown with different markers/colours, red circles mark the cluster means projected into 2D space, and red lines represent the two principal eigenvectors of the covariance matrices projected in 2D. (a) and (c) are non-angular analysis (b) and (d) angular analysis.
Visualisation of the yeast data with cluster labels, projected cluster means and projected eigenvectors of the cluster covariance matrices. Different cluster labels are shown with different markers/colours, red circles mark the cluster means projected into 2D space, and red lines represent the two Principal eigenvectors of the covariance matrices projected in 2D. (a) and (c) are non-angular analysis (b) and (d) angular analysis.
Covariance Matrix

The covariance matrix (inverse precision) provides information about the size and shape of a cluster in different directions in the higher dimensional space. Thus, small values in the covariance matrix suggest the genes in a cluster are very tightly grouped, with a high data density close to the mean, for a particular direction. Larger covariances suggest clusters with much more variation in expression/angular values. Although this information is very useful, it is not easy to visualise this information. One possibility is to sample points in the high dimensional space on the surface of a deformed hyper ellipse that reflects the shape of the covariance. These points could then be mapped into the 2D space. Alternatively, a grid of points in the high dimensional space could be sampled and their position and posterior probability in the 2D space to show how the projection onto the GTM manifold has deformed the probability space. However, sampling points in high dimensional space can be computationally inefficient and so these would not be useful in practice, particularly for very high dimensional data sets. An alternative therefore is to sample points along the eigenvectors of the covariance matrices and project these into the 2D plot. The eigenvectors of the covariance matrices are essentially the principal components of the cluster, and thus show the axes of maximum variability of the data, providing information about how tightly grouped the gene clusters are.

Figures 6.5 and 6.6 show the lines corresponding to the eigenvectors projected into 2D space. In order not to clutter the visualisation only the two eigenvectors with the two largest corresponding eigenvalues (first two Principal components of the covariance matrix) are shown. Points along the lines were equidistant in the data space, so the relative distances between the points in the 2D space indicates the deformation of the cluster once projected onto the manifold into 2D space. The angular GTM analysis of the yeast data shows two clusters with long lines representing the projected eigenvectors, suggesting either clusters with large covariances in some directions or clusters that have been modelled by multiple areas of the manifold. The majority of the clusters for the non-angular yeast visualisation show data distributed over large areas of the plot.
Table 6.1: Highly abundant PFAM annotations in cluster D. The hypergeometric distribution is used to test the null hypothesis that these abundances could occur at random in a cluster of 1027 genes. The entire set contained 13666 genes of which 2968 were used for clustering. Here, the p-value gives the probability of \( n \) or more genes in a subset of size 1027 having annotation \( i \), given that \( m \) genes are annotated with annotation \( i \) in the entire data set of size 13666.

6.3.2 Visualisation of Annotations

Protein Family Annotations (PFAM)

The protein family annotation for the majority of the genes in the rice blast data set are unknown, however, a few groups have been well annotated, for instance ribosomal proteins and endonucleases. Although these groups are not thought to have a role in the pathogenic behaviour of the fungus, and are not interesting in that sense, they can still be used to demonstrate the visualisation of groups of genes in the data set. It can be seen from figure 6.7 that the majority of the ribosomal genes are contained within the area of the plot associated with cluster C. In fact 60 genes annotated with ‘ribosomal’ are found within cluster C, and only 7 genes with this annotation are found in other clusters, which is quite striking. Similarly, there are other protein family annotations whose abundance in cluster D is far higher than expected. These are seen in figure 6.7, where the genes with these annotations are concentrated in the area of the plot associated with cluster D. The hypergeometric distribution can be used to test a null hypothesis that the abundances of these genes amongst the different clusters occur at random. The corresponding p-values for the annotations are shown in table 6.1 and are all less than \( 10^{-12} \), signifying the extremely low probability that these gene groupings would occur at random.
6.3 Visualisation of the Microarray Data sets

Figure 6.7: Rice blast data - visualisation of cluster labels and protein family information. (a) The four clusters discussed in chapter 5 following the angular distribution decomposition. (b) Ribosomal genes (according to PFAM annotation) are highlighted. Most appear in the area of the plot associated with cluster D. (c) PFAM annotations that are highly abundant in cluster D but not other clusters are highlighted (see table 6.1).
6.3 Visualisation of the Microarray Data sets

Figure 6.8: Rice blast data - visualisation of genes highlighted as potential pathogenicity factors in Dean et al. (2005). Many of the secreted proteins appear in areas of the plot associated with clusters A, and C - the clusters of genes showing up-regulation on one or both surface types during infection related development.

6.3.3 Visualisation of Selected Subsets

Potential Pathogenicity Factors

Whilst the examples shown in section 6.3.2 suggested tight groupings of genes with similar protein families, genes with the same annotation are not always expected to cluster together. There will be groups that are not tightly clustered and are instead distributed across many different expression patterns. Figure 6.8 highlights the group of potential pathogenicity factors discussed in Dean et al. (2005), their distribution can be observed throughout the data space and clusters.

Potentially Secreted Proteins

A group of potentially secreted proteins taken from the MGOS database are highlighted in figure 6.9. Secreted proteins are thought to play a vital role in the pathogenic behaviour of the rice blast fungus. This list of proteins have been found using the SignalP algorithm (Nielsen et al., 1997; Nielsen and Krogh, 1998) to predict which proteins are secreted and then the ProComp software to predict which of those secreted proteins are extracellular. Those present in the rice blast data set used for clustering are highlighted in the in the GTM visualisation, the majority seem to be within the areas of the visualisation plot associated with clusters A-D. Within each cluster, the predicted secreted proteins are
highlighted. Some of the secreted proteins are located in clusters A and B (see table 6.2), the clusters associated with more up-regulation on the infection inducing surface, notably 12.4% of the genes in cluster A and 9.2% of the genes in cluster B are predicted as being secreted proteins. There are also many secreted proteins in cluster C, a cluster associated with up-regulation on all surfaces compared with the spore sample. Although this cluster has similar up-regulation on both surface types, the secreted proteins are being more expressed during development and may be important in allowing the fungus to determine the surface type and triggering infection related development. Notably, the cluster associated with similar down-regulation in all four conditions contained the smallest percentage (1.5%) of secreted proteins.

**Yeast Environmental Shock Response Genes**

The distribution of the induced and repressed ESR genes can be seen in figure 6.10. In the non-angular approach the two different groups are well separated, but spread amongst multiple clusters. In the angular clustering and visualisation, the two groups are also well separated, but not distributed through the data space to the same extent.
Table 6.2: Distribution of the 739 genes predicted to correspond to secreted proteins using the SignalP and ProComp algorithms across the rice blast clusters. RMD is the group of genes removed prior to clustering as having less than 2-fold expression change in any of the conditions. Clusters A, B, and C associated with up-regulation during development contain a much higher percentage of secreted proteins than cluster D, associated with down-regulation.

Figure 6.10: Yeast data - visualisation of ESR gene groups. (a) Angular analysis (b) Non-angular analysis. The repressed ESR genes are highlighted in green and the induced highlighted in red. Blue crosses mark the centre of the variational Bayes clusters associated with the data set projected in the 2D plot.
6.4 Summary

The visualisation of gene expression data could be useful in highlighting structure or groupings of data points within a data set to provide a quick overview of the distribution of genes amongst expression patterns. The results of GTM can be difficult to interpret by themselves, even with the use of magnification factors. However, information from other sources such as the variational Bayes clustering can be incorporated into the visualisation.
to provide a more informative 2D map of the expression patterns, allowing the different areas of the plot to be associated with the underlying expression patterns found from clustering methods. Other information and selected subgroups of genes can also be easily highlighted, allowing the distribution of expression patterns within these groups of genes to be easily observed. Some selected gene groups tightly cluster, showing similar expression patterns, other selected groups show great variation in expression patterns, suggesting they are not controlled by the same regulatory proteins.
7

Locating Combinations of Transcription Factor Binding Sites: A Matrix Factorisation Approach

7.1 Introduction

The previous chapters have been concerned with the analysis of gene expression data in order to highlight interesting groups of genes that show similar expression patterns (by some definition of similarity) but what is a more interesting question is why these genes show similar expression patterns. Chapter 2 gave details of gene regulation and how gene expression can be controlled by the binding of proteins called transcription factors (TFs) which interact with the basal transcription apparatus and affect the rate at which transcription is initiated. Each TF has a specificity for binding certain DNA sequences known as transcription factor binding site (TFBS) motifs (or promoter sequences).

Most algorithms modelling gene regulation have concentrated on predicting the location of these TFBS in the regions upstream of transcription start site. These TFBS sites are modelled as short, highly conserved, sequences within longer less conserved regions. For some species transcription factor binding site motifs have been learnt from experimental data and algorithms have been developed to predict the presence of these known motifs. Motifs are often modelled as position weight matrices (PWMs), which are calculated by aligning all known examples of binding sites for a transcription factor and then calculating the frequency of each of the four nucleotides (A,C,G and T) at each position in the binding site (Stormo, 2000). The columns of the matrix can be normalised to give a probability of each nucleotide at a particular position.

A major problem with these methods is that they often give high false positive rates, where
the sequences occur at random rather than being a functional binding site. These false positives are by definition similar to those which the transcription factor is known to bind to but, without further information with which to model the activity of transcription factors, it is impossible to distinguish which are functional and non-functional sites. In practice the interaction of multiple transcription factors is often essential to the transcription of the gene. Therefore, attempts have been made to improve the specificity by modelling clusters of motifs rather than individual instances. In particular, algorithms have been developed that search in different windows of the sequences for dense clusters of motifs, known as regulatory modules, which have been found to occur in some promoter regions. Popular strategies for this have used logistic regression analysis (Krivan and Wasserman, 2001; Wasserman and Fickett, 1998), hidden Markov models (HMM) (Frith et al., 2001, 2002; Bailey and Noble, 2003) and combined p-value scores. The HMM approaches model each base in the sequence as being a part of a motif, as gaps between motifs in a module, or in the background sequence between modules. State paths are scored using log likelihood ratios and the significance for the score of the state path is then calculated using p-values or E-values (the expected number of state paths with a score greater than or equal this state path score that occur at random). These models can be effective when there is training data or prior knowledge about the modules such as the expected number of each motif and the gaps between the motifs, but this is not always available. Strategies using combined p-value scores search through different possible combinations and output those motifs combinations that are the most significant. Aerts et al. (2003) developed an algorithm to search for combinations of motifs occurring across multiple sequences, the algorithm used a combined p-value score and a heuristic graph search algorithm was used to search through the possible combinations. MScan (Johansson et al., 2003) takes different windows of a sequence and calculates a significance score for each motif within that window. It then looks for clusters of motifs (modules) with small p-values occurring in that window using a combined p-value score and outputs any modules that have a score below a specified threshold.
A transcription factor can affect the regulation of multiple genes and it is possible that co-regulated genes contain binding sites for the same transcription factors. This chapter presents a method of finding combinations of known (or previously predicted) TFBS motifs that occur in the promoter regions of a chosen group of genes. Unlike the methods described above the false positives are removed by looking for motif combinations that occur anywhere in the promoter regions of multiple genes rather than looking for dense clusters occurring in a specified window size. The model presented in this chapter looks for the highest scoring example of each motif in the input sequence for a gene. The maximum posterior probability of the motif given segments of the same width as the motif is used for this score. A matrix containing the scores for each motif in each sequence is constructed and then decomposed such that combinations of motifs occurring across multiple sequences are highlighted. The matrix factorisation projects the information in the motif score matrix into two other matrices using non-negative matrix factorisation. Thus one of these matrices provides information about a new set of variables, that are combinations of motifs (linear combinations of the original variables) and the second matrix specifies the genes in terms of which motifs combinations they are associated with. The matrix factorisation method can also be thought of as a de-noising algorithm, in which high motif scores that do not occur across multiple sequences are treated as noise (false positives) and effectively removed from the motif score matrix. It is shown, using simulated data, how the results can be visualised and interpreted.

7.2 Methods

7.2.1 Motif Score Matrix

Given a list of $M$ motifs and $N$ promoter sequences, an $M \times N$ motif score matrix, $S$, is defined such that the element $s_{ij}$ gives the score for the presence of the $i$th motif in the $j$th sequence. Each motif is specified by a position weight matrix, $Q$ such that $q_{b,a}$ is the probability of nucleotide $b$ being present at position $a$ in the motif sequence and where $\sum_b q_{b,a} = 1$. If $w_i$ is the width of the $i$th motif then
\[ Q^i = \begin{bmatrix} q_{A,1} & q_{A,2} & \cdots & q_{A,w_i} \\ q_{C,1} & q_{C,2} & \cdots & q_{C,w_i} \\ q_{G,1} & q_{G,2} & \cdots & q_{G,w_i} \\ q_{T,1} & q_{T,2} & \cdots & q_{T,w_i} \end{bmatrix} \]

The promoter regions for the selected genes are sequences of letters from the alphabet \(A,C,G,T\) and can be of equal or differing lengths. One possible score for the presence of motif \(i\) in sequence \(j\) as the maximum of the posterior probabilities for motif \(i\) given a segment of width \(w_i\) from sequence \(j\). Defining \(\text{Seg}_{ij}\) to be the set of segments of the sequence of width \(w_i\), and \(\text{Seg}^k_{ij}\) to be a member of this set we can calculate this score as

\[
S_{ij} = \max_k P(\text{Motif}_j | \text{Seg}^k_{ij})
\]

Using Bayes’ rule, \(P(\text{Motif}_j | \text{Seg}^k_{ij})\) can be written as

\[
P(\text{Motif}_j | \text{Seg}^k_{ij}) = \frac{P(\text{Seg}^k_{ij} | \text{Motif}_j)P(\text{Motif}_j)}{P(\text{Seg}^k_{ij})}
\]

Using a simple model to calculate the score for \(\text{Motif}_j\), it is assumed that each segment is either generated from the motif model or from a background model, \(B\), such that

\[
P(\text{Seg}^k_{ij}) = P(\text{Seg}^k_{ij} | \text{Motif}_j)P(\text{Motif}_j) + P(\text{Seg}^k_{ij} | B)P(B)
\]

The likelihood of a segment given the motif model can be calculated by multiplication of the relevant entries in the motif matrix. For instance if \(\text{Seg}^k_{ij}\) is a sequence of bases \(b_1, b_2, ..., b_{w_i}\), then

\[
P(\text{Seg}^k_{ij} | \text{Motif}_i) = \prod_{a=1}^{w_i} q_{ba,a}^i
\]

**Background models**

A third order Markov model of intergenic regions is used for the background model as this has been previously shown to be superior to a zero order model of relative base frequencies.
However, in practice, any background model thought appropriate could be used for the calculation of \( P(\text{Seg}^k_{ij} | B) \). The third order model is given by

\[
P(\text{Seg}^k_{ij} | B) = \prod_{a=1}^{w_j} P(\text{b}_a | \text{b}_{a-1}...\text{b}_{a-3})\]

The three bases in the promoter sequence that proceed the Seg\(^k_{ij}\) can be used for the calculations involving \( b_0, b_{-1} \) and \( b_{-2} \).

**Priors**

Considering a sequence of length 1000bp, then there will be \( 1000 - w_j \) different segments that could correspond to motif of width \( w_j \), however it is expected that Motif\(_j\) will only be found a small number of times at most in the upstream promoter region of a gene. The other segments will not be matches to the motif and the probability that they came from the background model should be larger than the probability that they were generated from the motif. Thus, the priors were set to \( P(\text{Motif}) = 1/500 \) and \( P(B) = 1 - P(\text{Motif}) \) for all motifs.

**7.2.2 Decomposition of the score matrix using non-negative matrix factorisation**

The matrix \( S \) contains maximum posterior probabilities for the motif given the segments in a sequence, so all entries in the matrix will be non-negative and lie in the range [0,1]. Non-negative matrix factorisation (NNMF) (Lee and Seung, 1999, 2001) can be used to find latent structure in non-negative matrices and it is shown here how NNMF can be used to find combinations of motifs present in the promoter regions.

Given the \( M \times N \) scoring matrix, NNMF factorises the matrix into two further matrices, \( U \) and \( V \) of size \( M \times R \) and \( R \times N \) respectively, such that

\[
S \approx UV
\]

and where all elements of \( U \) and \( V \) are in \( \mathbb{R}^+ \). If a good approximation to \( S \) is found with \( R < N \) then there is latent structure in the data. \( U \) can be thought of as containing a basis for \( S \) that provides the optimised linear
7.2 Methods

Figure 7.1: NNMF can be used to decompose the motif scores into information regarding combinations of motifs that occur in the sequences and which sequences these motif patterns are present in.

Figure 7.2: Visualisation of a simple example of the $U$ and $V$ matrices, showing how the patterns can be interpreted when applied to DNA sequences from promoter regions of genes. Black corresponds to a large value (high probability of motif pattern), and white is small value (no pattern).

approximation of the data in $S$. In the case of the motif score matrix, $U$ corresponds to a basis of motif combinations, specifying which motifs are present in which combination. Each column of $V$ corresponds to a promoter sequence and the values in the $jth$ column of $V$ give a weighting for the presence of the each combination of motifs being present in sequence $j$. NNMF can therefore be used to decompose the motif scores into information regarding combinations of motifs that occur in the sequences and which sequences these motif patterns are present in (see figures 7.1 and 7.2).

$U$ and $V$ can be calculated using an iterative update procedure which is simple to implement (Lee and Seung, 1999, 2001). The aim of the iteration is to minimise the error
between the approximation given by $UV$ and the true matrix $S$. The sum of square errors is calculated as

$$f = \sum_i \sum_j (s_{ij} - (UV)_{ij})^2$$

Given an initial random guess at $U$ and $V$, each element can be updated by a factor proportional to the amount of error in the current approximation until the error converges to a local minimum. The update equations for the elements of $U$ and $V$ are:

$$v_{ir} \leftarrow v_{ir} \frac{(U^T S)_{ir}}{(U^T UV)_{ir}}$$

$$u_{rj} \leftarrow u_{rj} \frac{(S V^T)_{rj}}{(U V V^T)_{rj}}$$

### 7.3 Results on Simulated Data

DNA sequences were simulated for the promoter regions of the yeast *Saccharomyces cerevisiae* using PWMs available from the *Saccharomyces Cerevisiae* Promoter Database (SCPD) and the TRANSFAC database. Sequences of 500bp were generated from a 3rd order Markov background model with transition probabilities calculated from the intergenic regions of the *Saccharomyces Cerevisiae* genome. Transcription factor binding site sequences were then generated from selected motif models and these motif sequences inserted into the background sequences at certain positions.

As some of the PWMs were calculated from only a small number of experimental examples, many of the entries in the PWMs were zero. This could be because the sample size was too small to provide an accurate representation of the TFBS sequences. Therefore, a small amount ($1 \times 10^{-10}$) was added to all the entries in the PWMs and they were re-normalised. The motif score matrix was then calculated for each of the simulated data sets using the entire set of 43 motifs. These matrices were then decomposed using non-negative matrix factorisation with various values of $R$. 
Ideally the motif scores would be high when there is a good match to the motif in the promoter region. Matches can occur because there is a segment that is a functional transcription factor binding site, or a segment that may have occurred at random and not be functional as a transcription factor binding site. Given only knowledge of the motif matrix and the a model of how frequently sequences occur in the intergenic regions it is impossible to distinguish true positives and false positives in these cases. Taking the hypothesis that combinations of TFBS are important, rather than individual instances, some of the false positives can be removed because randomly occurring (non-functional) matches are less likely to occur in the same combinations across multiple promoter regions. However, for some motifs, the distribution of scores for true positives is similar to the distribution of false positives because the sequences occur frequently at random from the background distribution. This a potential problem as the score for these motifs could appear as structure in the motif score matrix. To demonstrate this a data set was created consisting of 100 sequences that had had all the motifs inserted and 100 sequences that had had no motifs inserted. A motif that is distinguished easily from the background model therefore should have high scores for the first 100 sequences and low scores for the second 100 (see figure 7.3). We can see from this that five motifs in particular have a high false positive rate, and the distribution of false positive scores is similar to the true positive scores (figure 7.4). This problem will affect any algorithm using a significance score based on the match of the segments to the motifs modelled only with PWMs. In some sense, it is not incorrect to predict the presence of these motifs, as there must be good matches to the motif in order to give the high scores, and some of them could be functional. However, it is useful to know which of the motifs occur frequently at random as the scores for these binding sites could dominate the motif matrix and appear as a motif combination in the matrix factorisation results. Biologically, this means that it is possible that these sites could be functional in the promoter regions of the genes but it is predicted with a smaller posterior probability than for other motifs. Further simulated data sets will be used to show how this NNMF approach deals with this problem.
Figure 7.3: Visualisation of motif scores for 200 simulated sequences. Sequences 1-100 have had all 43 motifs artificially inserted, sequences 101-200 are generated only from background model and have had no motifs inserted. For many of the motif patterns, the distribution of motif scores for the first 100 sequences is very different to that for the second 100 sequences. These motifs are easily distinguished from background sequences and have relatively low false positive rates. Some of the motifs have overlapping distributions of motif scores and a higher false positive rate. These false positives are unlikely to occur in the same combinations in each sequence and therefore can be removed using the non-negative matrix factorisation approach. For some motifs, five in particular, the distributions of scores are quite similar, with the mean score from both being quite similar. These motifs have high false positive rate, appearing frequently from the background model, and may appear as a potential motif combination in all results.
7.3 Results on Simulated Data

Figure 7.4: Distributions (as histogram plots) of the motifs scores for some of the motifs shown in figure 7.3. Black bars show the distribution for scores of positive examples and white bars show the distribution of scores for negative examples. Motifs 1, 3, 8 and 9 have similar or overlapping distributions, making it difficult to distinguish true positives and false positives based only on motif score. However, many of the motifs, such as 22, 28, 33, 40, 41 have much more distinct distributions with the majority of true positives having high motif scores (close to 1.0) and true negatives having low motif scores (close to 0.0).

7.3.1 Data sets

6 different TFBS motif combinations were chosen of the 43 available in total from SCPD and TRANSFAC to insert into the sequences (see table 7.1). Data sets were therefore constructed by generating sequences containing different motif combinations and the results for two of these data sets (see table 7.2) are presented here. To demonstrate how multiple copies of the same motif can be identified using this method, each motif is represented by

<table>
<thead>
<tr>
<th>Combination</th>
<th>TFBS</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>TBP, Matalpha2 (x2), MCM1, STE12</td>
</tr>
<tr>
<td>C2</td>
<td>TBP, Matalpha2, SCB (x2), MCB</td>
</tr>
<tr>
<td>C3</td>
<td>TBP, REB, GCN4 (x2), MCB</td>
</tr>
<tr>
<td>C4</td>
<td>TBP, RLM1, repressor of car, ABF1.01</td>
</tr>
<tr>
<td>C5</td>
<td>TBP, SMP1, HSF.03</td>
</tr>
<tr>
<td>C6</td>
<td>TBP, SMP1, HSF.03, RLM1, repressor of car, ABF1.01</td>
</tr>
</tbody>
</table>

Table 7.1: Six motif combinations were selected for insertion into the simulated sequences. Some combinations contain two copies of the same motif.
7.3 Results on Simulated Data

Figure 7.5: Visualisation of the correct results for the different motif combinations, using similar representation as shown in figure 7.2. Note that there are only five combinations as combination C6 (table 7.2) is a linear combination of C4 and C5.

<table>
<thead>
<tr>
<th>Data set</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>No motif combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>30</td>
<td>20</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>35</td>
<td>30</td>
<td>20</td>
<td>15</td>
<td>15</td>
<td>25</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 7.2: Data sets were constructed as sets of sequences containing the different motif combinations shown in table 7.2. For example, data set 1 contains 90 sequences in total. 25 of them contain C1, 30 contain C2, 20 contain C3 and the final 15 sequences have had no combinations inserted.
two rows, the first being the highest score for the motif and the second row being the second highest score for the motif in each sequence. Therefore, the motif score matrix contained 86 scores for each sequence. The correct motif combinations used in the simulated data sets are shown in figure 7.5. The matrix factorisation results for the data sets are shown in figures 7.6 and 7.7. The results on both data sets identified the correct combinations and the true structure was highlighted in the reconstructed (de-noised) version of the motif score matrix.

**Decomposition into patterns**

Figures 7.6 and 7.7 show the decomposition results for different values of $R$. The results of the data set containing three combinations of motifs shows the correct motifs have been identified. With $R = 3$ those motifs with a high false positive rate also occur, but are much weaker than the true motifs. If $R = 4$ the motifs with a high false positive rate appear as an extra combination, but the true motif combinations are still correctly identified. Similarly, the results for the data set with six different combinations inserted correctly identify the underlying motifs and that the fact that sixth combination is a mixture of two of the others. Like the previous data set the motifs with high false positive rate are predicted but with much lower probability than the true motifs. Therefore, provided there is sufficient structure in the motif matrix, in the sense that the subset of genes under study do share similar transcription factor binding sites, this method is able to locate this underlying structure and the results are not dominated by the frequently occurring motifs.

**Multiple copies of motif**

In some cases it is known that multiple copies of the same transcription factor bind to the promoter regions of a gene. Therefore, there will be multiple binding sites for this transcription factor. We have demonstrated how this technique can highlight these by using the scores for the two highest scoring segments. If there are multiple copies of the same motif then high scores should be seen for both. In combinations C1, C2 and C3, two copies of the same binding site were inserted and the results show the two copies are
correctly identified in the motif combinations.

**Multiple motif models for the same TFBS**

Some transcription factors had multiple PWMs, either taken from different databases or submitted from different experimental results. Some of the extra structure occurring in the motif matrix corresponds to a TFBS that have been inserted into the simulated data, but a slightly different model. For instance, there are two PWMs for the binding site of Matalpha2, corresponding to positions 27 and 33 in the motif score matrix. The PWM from the SCPD database was inserted into the sequences but the PWM from TRANSFAC is also predicted in the motif combinations. As multiple very similar binding sites will be predicted in combination, methods which search for combinations within a particular window and use a combined p-value score to calculate significance, could incorrectly show these as being a highly significant combination. Alternatively, the constraint of non-overlapping binding sites could be used, and although most binding sites found so far are non-overlapping, overlapping binding sites have been observed in some cases, in which the rate of transcription is affected by the binding of one factor blocking the binding of another. As the number of transcription factor binding sites and organisms for which experimental information is available is relatively small, a model which makes the smallest number of assumptions is arguably preferable. The matrix factorisation approach is not vulnerable to problems associated with multiple copies (or very similar) binding site models and does not need to impose constraints about the relative positions of the binding sites.

**Sequences with no motifs**

In both data sets some sequences had had no motif combinations inserted, these can be seen by examining the results for $V$, where the weighting for combinations in these sequences is very small. The reconstructed matrix also highlights these, as no structure is present in the corresponding columns of the matrix.
### Table 7.3: Data set 2 results: the top five motifs associated with each combination.

#### Thresholding the reconstructed matrix

To further highlight the true structure within the motif matrix and remove noise, the reconstructed matrix can be thresholded, such that anything below a threshold value is set to zero. Examples of this can be seen in figure 7.8.
Figure 7.6: Data set 1: decomposition results for different values of $R$, showing $U$, $V$ and the reconstructed (de-noised) $U \times V$ matrix. The data set contained 3 different motif combinations. The final 15 sequences contained no motif combinations. Good results can be seen with $R = 3$, or $R = 4$. The result for $R = 4$ contains an extra combination, corresponding to those with high false positive rates, but the true combinations have been correctly found. In the result for $R = 3$ they appear together with the true combinations, but with lower scores than the true motifs. The extra motif predicted in the combinations corresponds to a second motif model for the Matalpha2 binding site, as this is similar to the Matalpha2 motif model that was inserted into the sequences, then high scores for this motif appear as well.
Figure 7.7: Data set 2: decomposition results for different values of $R$, showing $U$, $V$ and the reconstructed (de-noised) $U \ast V$ matrix. The data set contained five different motif combinations. The final 20 motifs contained no motif combinations and 15 of the sequences contained 2 of the combinations. The extra motifs found to be part of the predicted combinations (compare to figure 7.5) are because there are motif models similar to those that have been inserted.
Figure 7.8: Data sets 1 and 2: Thresholding of the de-noised matrices. To further highlight the true structure, the de-noised matrices can be thresholded, setting to zero any elements that are below a chosen threshold value. Shown for both (a) data sets 1 and (b) 2 are, from left to right, the original score matrix \((S)\), the reconstructed matrix \((UV)\) and the reconstructed matrix thresholded at a value of 0.5.
7.4 Results on Rice Blast Data

In the previous chapters it was shown how the analysis and visualisation of the rice blast data set identified patterns of expression change and groups of co-regulated genes. It would be of interest to know if these groups of co-regulated genes share common transcription factor binding sites, if there is a single binding site or combination of binding sites that seem responsible for the differential expression; or if there are multiple combinations of binding sites that are associated with the same expression pattern (when observed across the particular experimental conditions in the data set); or if there seems to be no common motifs. To do this a list of binding site motifs for the rice blast genome is required and then the matrix factorisation approach described in this chapter can be applied to each cluster of genes to search for the presence of a single combination, multiple combinations or alternatively show that no clear patterns can be found within the sequences. Unfortunately, not much is known about binding sites in the rice blast genome so a list of position weight matrices can not be gained directly from experimental data. Thus in order to demonstrate the method on real data, the MEME algorithm (Bailey and Elkan, 1994) was used to predict potential binding sites for the gene sequences. MEME is one of a number of algorithms that look for small sub-sequences that appear multiple times within a set of promoter sequences. The NNMF algorithm can then be used to look for combinations of these individual sites predicted from MEME, which would be difficult to decipher from different MEME results manually. The motif score matrix can also be used to observe the similarities and differences in motif predictions from the different clusters, to see if combinations that seem to be associated with one cluster appear in any other cluster and if groups of genes that seem to be co-regulated do share common motifs. The technique is demonstrated on the rice blast data set to attempt to answer three questions regarding the clusters of genes identified in previous chapters. These are: (1) Are there any motif combinations specifically associated with clusters A or B? (2) Do the groups of genes in cluster D discussed in section 6.3.2 share the same motifs? (3) Are there any similarities or differences in the promoter regions of the secreted proteins?
7.4 Results on Rice Blast Data

7.4.1 Are there any motif combinations specifically associated with clusters A or B?

It was noted in chapter 5 that clusters A and B from the angular distribution decomposition analysis are associated with up-regulation on a hydrophobic surface, and therefore some of the genes within these clusters could be controlled by regulatory pathways involved in pathogenic behaviour. MEME was run on sequences 1000bp upstream of the transcription start sites for the genes in these two clusters to predict potential motifs. The parameters of MEME were set to search for motifs of width 5bp-15bp and to output the 10 highest ranking motif predictions for each cluster. The motif score matrix was then calculated using these predicted PWMs and the two highest ranking scores of each motif in each sequence were recorded so that any patterns of multiple copies of a motif could be extracted by the NNMF analysis.

On first inspection of the motif score matrices for the clusters (figures 7.9 and 7.10), it can be seen that some of the motifs do appear across a large proportion of genes within the clusters, whereas some of the motifs do not seem to appear across many genes in the cluster or in a particular combinations. The NNMF can be used to extract those sites that appear frequently and in combinations. Figures 7.9 and 7.10 shows the decomposition results for 5 motif combinations in cluster A and 4 motif combinations in cluster B, however most of the ‘combinations’ actually only include one motif, showing that these motifs actually seem to appear separately in the sequences rather in combination with others. Inspection of the PWMs for the motifs within combination 1 for cluster A and combination 1 for cluster B shows that the motifs are all describing T-rich sequences (appendix D), so they appear as a combination because they are in fact describing very similar motifs.

The next question to answer is whether the motifs associated with these clusters occur specifically within these clusters or if they also occur frequently in clusters showing very different expression patterns. This can again be done using the results of the NNMF method quite efficiently, by calculating the motif score matrices for the motifs in promoter regions of genes within other clusters and then projecting these matrices into the variables
Figure 7.9: Rice blast data - visualisations of the NNMF analysis for the motif predictions and sequences for 1000bp upstream of genes in cluster A. $S$ : Motif score matrix, the rows represent the two highest scores for the 10 motifs predicted from MEME for cluster A. The columns represent the genes in cluster A for which 1000bp upstream sequence data was available. $U \ast V$ : The approximation to the motif score matrix with five motif combinations. $U$ : The matrix describing the motifs associated with each of the combinations. Combination 1 appears to contain two of the motifs from the MEME predictions and multiple copies of the motif are found, the other ‘combinations’ only contain single motifs. $V$ : matrix showing which of the five combinations are associated with each sequence. Combination 1 occurs in the upstream regions of most of the genes, the others do not occur as frequently or in any particular combinations.
Figure 7.10: Rice blast data - visualisations of the NNMF analysis for the motif predictions and sequences for 1000bp upstream of genes in cluster B. \( S \): Motif score matrix, the rows represent the two highest scores for the 10 motifs predicted from MEME for cluster B. The columns represent the genes in cluster B for which 1000bp upstream sequence data was available. \( U^*V \): The approximation to the motif score matrix with four motif combinations. Combination 1 contains three of the motifs, all of them in multiple copies. \( U \): The matrix describing the motifs associated with each of the combinations. \( V \): matrix showing which of the four combinations are associated with each sequence. Combination 1 occurs in the upstream regions of most of the genes, the others do not occur as frequently or in any particular combinations.
7.4 Results on Rice Blast Data

describing the motif combinations in clusters A and B, to see if these combinations are present. Figures 7.11 and 7.12 show that these motifs do appear quite frequently in the other clusters, both those up-regulated and those down-regulated and so cannot explain the differential regulation of genes within these clusters. Therefore, the search for any motif combinations that may explain the particular expression patterns associated with clusters A and B must continue. It is perhaps not surprising that this motif is commonly occurring, as regions with high AT content (such as the TATA box) are known be found in promoter regions because the DNA strands are much easier to unravel in AT-rich regions, which is required for transcription to occur (Calladine et al., 2004). There are many reasons why only motifs that occur commonly across sequences in the rice blast genome have been highlighted by the MEME analysis. It may be that motifs not contained with the regions 1000bp upstream of the transcription start site are involved in the regulation, so different sequence windows need to be searched. Perhaps there are multiple combinations of motifs found within the clusters, so none of the combinations are found in the majority of the sequences, and hence the signals were not strong enough for the MEME algorithm to locate. It is also possible that there are actually no motifs associated with groups of genes in these clusters.

One final point of interest with reference to the motifs predicted for cluster A is that of the genes in cluster D labeled with the “RNase H” pfam annotation. Although it seems that many genes in cluster D do contain these commonly occurring T-rich motifs, the “RNase H” genes notably do not (figure 7.13). However, they do contain a different combination of the motifs actually predicted from cluster A.

7.4.2 Do the groups of genes in cluster D discussed in section 6.3.2 share the same motifs?

Section 6.3.2 discussed five groups of genes within cluster D, distinguished by their PFAM annotations, but that were all associated with the expression pattern of down-regulation at both points and on both surface types post inoculation. The motif matrix and NNMF analysis can be used to discover whether these groups contain similar motifs in their pro-
Figure 7.11: Rice blast data. Motif score matrices and projection on the five motif combinations for motifs in cluster A shown in figure 7.9 for the 1000bp upstream regions of genes in (a) cluster B, (b) cluster C and (c) cluster D. Motif combination 1, which is found frequently occurring in the upstream regions of cluster A, is also commonly found in the upstream regions of the other clusters.
7.4 Results on Rice Blast Data

Figure 7.12: Rice blast data. Motif score matrices and projection on the five motif combinations for motifs in cluster B shown in figure 7.10 for the 1000bp upstream regions of genes in (a) cluster A (b) cluster C and (c) cluster D. Motif combination 1 is also commonly found in the upstream regions of the other clusters.
7.4 Results on Rice Blast Data

Figure 7.13: Rice blast data. (a) The motif score matrix for the genes associated with pfam annotation “RNase H” and set of the motif predictions from cluster A. (b) The projection of the motif score matrix onto the five motif combinations from the NNMF analysis visualised in figure 7.9. Although motif combination 1 occurs frequently in upstream regions of genes, notably it does not occur in the upstream regions of the “RNase H” genes. A different combination of motif predictions is however present.

moter regions, or whether they are controlled differently and just happen to show similar expression profiles across the small number of conditions under study. This example also demonstrates how patterns can be analysed across both upstream and downstream regions simultaneously rather than requiring a single continuous sequence region for each gene, and how motif predictions from multiple groups can be used in one large motif score matrix to observe the similarities and differences in motif combinations present across the different groups. MEME was again used to predict motifs for each of the groups individually, and the 10 highest ranking for each group output, giving a total of 50 motifs in the upstream regions. MEME was also used to predict motifs in the 1000bp downstream regions of these genes, and the 10 highest ranking were chosen, providing 50 motif predictions for downstream regions. The motif matrix was scored on all motifs across all the sequences with promoter sequence available in these five groups, and the two largest scores for each motif in each sequence recorded.

Figure 7.14 visualises the motif matrix and matrix decomposition results for motifs predicted by MEME. Motifs 1-100 are the scores for the motifs predicted from the upstream regions and motifs 101-200 are the motif predictions from the downstream regions. Perhaps unsurprisingly, the results showed that the gene groups contained that motifs that
7.4 Results on Rice Blast Data

Figure 7.14: Rice blast data - visualisations of the NNMF analysis for the motif predictions and sequences for 1000bp upstream and 1000bp downstream of genes in five Pfam groups. (a) Motif score matrix, the rows represent the two highest scores for the motifs predicted by MEME analysis (10 from upstream regions and 10 from downstream regions per group). G1=PF03184 “DDE superfamily endonuclease”, G2=PF00078 “Reverse transcriptase (RNA-dependent DNA polymerase)”, G3=PF00665 “retroviral pol related endonuclease”, G4=PF00075 “RNase H”, G5=PF03732 “Retrotransposon gag protein”. (b) The approximation to the motif score matrix with seven motif combinations. (c) The matrix describing the motifs associated with each of the combinations. (d) matrix showing which of the seven combinations are associated with each sequence. There are two different motif combinations associated with the promoter sequence in G1 (combinations 3 and 6) in (c). The analysis of 1000bp up- and downstream shows motifs that are largely associated with a single Pfam group, but the downstream motifs for G2 and G4 seem to be the same, forming combination 2. There are also a few genes that do not have the motif combination associated with their Pfam group in their promoter regions, but do contain a combination associated with one of the other Pfam groups.
Figure 7.15: Rice blast data. (a) Motif score matrix for the motifs predicted for the pfam groups in cluster D in the upstream and downstream regions of the other genes cluster D. (b) The motif score matrix projected onto the seven combinations for the pfam annotations shown in figure 7.14. The visualisation shows that the combinations also appear in the promoter regions of other genes in cluster D, not just those that have been already associated with the pfam groups.

had been predicted for those genes groups by MEME. However, the motif score matrix and NNMF analysis allows other information to be seen that is not as easily deciphered directly from the results of prediction methods such as MEME. For instance, they show that there are two different combinations present in group 1, and that some of the motifs predicted for the downstream regions of group 1 were only found in a few sequences, so did not appear in the combinations and were treated as noise. It can also be seen that although the upstream predictions for groups 2 and 4 were different, the two groups appear to contain the same motifs in the downstream regions. Figure 7.14 also shows that some of the sequences did not contain the motif combination associated with their pfam group, but instead did contain a combination from one of the other groups, so while in general each pfam group has a different combinations of motifs in regions 1000bp upstream and downstream, there does seem to be some cross “contamination” of the motif combinations between the groups.

The motif score matrix for the other genes in cluster D was projected into these motif combinations to search for any other genes that contain these sequence in their upstream or downstream regions. The results of this projection can be seen in figure 7.15, showing that a few examples of these combinations occur in cluster D.
7.4 Results on Rice Blast Data

7.4.3 Are there any similarities or differences in the promoter regions of the secreted proteins?

In this third example applying the motif matrix factorisation analysis to the rice blast data, the promoter regions of genes corresponding to secreted proteins predicted by signalIP (see section 6.3.3) are examined. MEME was used to predict motifs for the secreted proteins within each cluster separately, then the NNMF was used to identify any commonly occurring combinations of motifs. The motif predictions for clusters A, C and D seem to be motifs that occur across all clusters (figure 7.16), and are therefore probably not of any value to explain the differential expression of the secreted proteins. However, for the other, smaller groups, MEME does seem to have identified motifs that occur in combination within the clusters and are not found in combination within other clusters. For example, those motifs predicted for the secreted proteins of cluster B seem to occur in combination in most of the sequences, but do not appear in the same combination in other clusters, and may therefore be specific motifs involved with the regulation of these genes. The same pattern can be seen for the secreted proteins in clusters E-H. It can also been from the figure some of the motifs predicted for the secreted proteins in cluster I are also found in some three of secreted proteins assigned to cluster D, although the combination of the motifs for is different to those in cluster I.

Next, the motifs score matrix for the promoter regions of all genes in cluster B was calculated, rather than just those predicted as secreted (figure 7.17). As expected the motifs from secreted proteins in clusters A, C and D, occur commonly across this cluster as they are probably motifs that occur frequently throughout the genome and not involved with differential expression. What is notable is that the motif combination associated with the secreted proteins of cluster B is not found in the promoter regions of any other genes in cluster B, so this motif combination seems to be localised to the promoter regions of the secreted proteins. The motifs in this combination may therefore be part of regulation pathway for just the secreted proteins, or it may be that these motifs just happened to appear randomly in the regions upstream of the eight secreted proteins that MEME searched for
commonly occurring motifs in.

It is important to note that the predictions from MEME, and other algorithms that predict binding site motifs, are sensitive to set of sequences, parameters and sequence lengths on which they are run. Therefore, although the analysis has highlighted motif patterns, which could be involved in the regulatory pathways, the motifs in these regions may just form part of the regulatory control, or indeed not be involved in regulation at all. It is therefore important to verify these motifs with the results of experimental investigation before any definite biological conclusions can be drawn.

7.5 Discussion

Presented here is a method that can be used to find combinations of TFBS motifs common to the promoter regions of multiple genes. The group of genes under examination can be chosen from various sources, such as gene expression analysis, or those with similar functions or annotations. It has been shown using simulated data examples how the underlying patterns can be extracted using NNMF by decomposition into two matrices. The first of these provides information about which motif combinations are present and the second shows which combinations occur in the sequences. A de-noised version of the score matrix can be constructed and thresholded to highlight the underlying components.

The major drawback of this method is the current quality of the binding site models. Only a small number are known and they are based on a small number of experimental samples. This is a problem affecting all algorithms that attempt to locate known motifs. Therefore, this method is likely to prove more useful in the future when more experimental data is available or the motif prediction algorithms provide high quality predictions, but it has been shown how the technique could be applied to real data to highlight patterns across different groups.

Although the analysis of TFBS in this chapter has concentrated mainly on locating motifs in the regions (500bp-1000bp) upstream of the TSS, this method could also use sequence information from other regions as well. Rather than requiring a continuous sequence of
Figure 7.16: Rice blast data - visualisations of the NNMF analysis for the motif predictions and sequences for 1000bp upstream of genes corresponding to secreted proteins. (a) Motif score matrix, the rows represent the two highest scores for the 10 motifs predicted from MEME for the secreted proteins in each cluster. $m - X$ indicates the motifs predicted for the secreted proteins in cluster X. The columns represent the genes for which 1000bp upstream sequence data was available. A-I indicate the clusters from the angular distribution decomposition analysis that the gene encoding the secreted proteins were associated with. (b) The approximation to the motif score matrix with seven motif combinations. (c) The matrix describing the motifs associated with each of the combinations. (d) Matrix showing which of the seven combinations are associated with each sequence. The predictions for clusters A, C, and D appear commonly in the promoter regions of genes in all of the clusters. The motifs predicted for the promoter regions for clusters B and E-H do not appear in the same combination in other clusters. The motifs predicted for cluster I also seem to occur in the promoter regions of three of the secreted protein genes in cluster D but none of the other other clusters.
Figure 7.17: Rice blast data. Motif score matrix for the motifs predicted from the groups of secreted proteins in the upstream regions of genes in cluster B. The motifs predicted from the secreted proteins in clusters A, C and D, occur commonly across in the upstream regions of genes in cluster B. The motifs combinations for the 8 secreted proteins in cluster B can also be seen, but it is notable is that this motif combination is not found in the promoter regions of any other genes in cluster B.

a specific length, multiple regions can be examined for a gene and a different set of regions can be examined for each motif. For example, proteins also bind to DNA sequences (enhancers) much further away from the TSS and control transcription by altering the position of the genes on the histones. Although these enhancer sequences are found much further away from the TSS than the TFBS’ (either upstream or downstream) they can have a dramatic effect on the transcription of a gene. Therefore, if information about possible enhancer sequences and their possible locations is available for particular genes, these parts of the genome could be considered in the motif score matrix as well. It is also straightforward to include the complementary or reverse sequences when scoring the presence of potential binding sites in the sequences.

It is not known whether transcription binding sites for all regulatory mechanisms occur in dense clusters (modules). The complex folding and structure of DNA could mean that it is the relative positions of the binding sites in the 3D structure of the chromosome rather than the positions within the sequence of bases that are important for the transcriptional regulation. In this approach, no assumption is made that the proteins which affect transcription have to be close together, the algorithm simply finds combinations of potential
sites that could affect the transcription of a gene.

Secondly, although this chapter has discussed combinations of TFBS, the method could also be applied to protein domains. Given a set of potential protein domains, the protein sequences for a group of proteins could be analysed to find similarities in the functional sites that they contain.
8.1 Summary

Understanding gene expression and gene regulation is vital to the understanding of molecular biology and the differences in the molecular interactions occurring in a cell during disease and development. The development of high-throughput methods has lead to an explosion in the amount of data concerning gene expression and regulation in various organisms. These data sets contain valuable information but because of the large amount of data it is impossible to make sense of the data and extract the useful information hidden within it manually. Methods that can automatically locate or highlight patterns within these large, noisy, multi-dimensional data sets are greatly needed. The theme of this thesis is thus to investigate the use of different data projection techniques that can automatically discover and highlight patterns hidden within data. Projecting the data into different data spaces, which are defined by a new set of variables, can be useful if the the new set of variables are more meaningful for interpretation of the data or the size of the new set of variables is smaller.

In particular this thesis has focused on three main areas. Firstly, the automated analysis of gene expression data by using data transformations and clustering techniques which project data into a space which defines their association with a set of clusters. The patterns extracted from this type of analysis are therefore the set of clusters and the expression patterns they correspond to. Secondly, projection of gene expression data into a 2D space which allows visualisation of the data and the pattern extraction can then be done by the human. Multiple sources of information can be incorporated to aid the visualisation of gene groups. Thirdly, the use of matrix factorisation techniques to locate combinations of
transcription factor binding sites motifs. In addition it has shown how mixture modelling can be used with circular data. The developments in these areas will be discussed in more detail in the following sections.

### 8.1.1 Analysis of Gene Expression Data

Chapter 3 reviewed some data projection techniques and discussed how some of them had been previously applied to the problem of automated pattern finding in gene expression data. The techniques differ in their assumptions about the data, their parameterisations and their model order selection. Different transformations of the data change the distances between data points, allowing different definitions of similarity to be represented. Ideally, the data transformations and projections should project the data into a representation from which information about the associated expression changes can be inferred. However, in practice inferring the associated expression changes can be difficult, particularly after the application of transformations such as standardisation or if clusters are found in which the expression values show large variation.

Probabilistic modelling, such as mixture modelling, provides explicit handling of uncertainty about which data points are associated with which mixture component and, through the covariance parameters, provide information about the variance of the clusters across the different dimensions. Furthermore, Bayesian mixture modelling also handles the uncertainty in the inferred model parameters. Variational Bayes mixture modelling using the mean field model provides a computationally efficient method of inferring the parameters of the mixture model, in which the negative free energy maximisation naturally penalises over complex models as well as having the advantages of the Bayesian approach that provides information about the uncertainty in the model parameters and cluster labels.

Chapter 4 investigated the structures present in the data sets by looking at the pairwise comparisons of the data. In some pairwise comparisons, there is no relationship between the expression in the two conditions and the angular distribution of the data almost uniform, however comparisons of other conditions showed angular structures. These structures correspond to multiple genes showing the same changes in expression between two
conditions and it was hypothesised that there may exist clusters of genes that show similar angles across a series of comparisons. Thus, chapter 5 defined a method of locating any such clusters using a series of two data projections. Firstly, the log fold change data is transformed to angular data by taking a series of pairwise comparisons, then a variational Bayes clustering technique projects the data points into posterior probabilities for the different clusters. As the data is angular, the variational Bayes mixture model is also modified to work with circular distributions. It has also been shown how the expected values of the model parameters can be used to provide some automatic information about the clusters in terms of pairwise comparisons of conditions which can be useful in allowing biological researchers to locate the clusters of most interest to their investigation. An example of this for the rice blast data set is given in appendix C.

8.1.2 Visualisation of Gene Expression Data

Visualisation of gene expression data has been mainly done through the use of heatmaps or plotting of expression profiles. However, for large data sets or large clusters it can be difficult to see the different clusters in the heatmaps, or the similarities and differences between different clusters when plotting expression profiles for each cluster separately.

The idea of representing the expression data for all the genes on a 2D map was introduced in chapter 3, and so dimensionality reduction methods were discussed in terms of how they could project multi-dimensional data into 2D space. Chapter 4 investigated the techniques on simple data sets and demonstrated the differences in the projections obtained due to the different underlying assumptions of the models. The GTM algorithm was shown to be less sensitive to parameterisations than LLE and unlike PCA could model non-linear data. It is also able to cope with clustered data, despite the model assuming the data lies on a manifold. However, the proportions of the 2D map the clusters are projected over may not accurately reflect the size of the clusters in the original data space and the inter-cluster separation between different clusters will not be well modelled, so care must be taken as to which aspects of the visualisation are interpreted.

Chapter 6 showed some GTM visualisations of the microarray data sets. To aid the vi-
sualisation of gene groupings, heatmaps are often used in conjunction with a hierarchical clustering tree and expression profile plots can used with a k-means clustering result, in which the profiles for each cluster are plotted separately. In a similar way, the GTM visualisation can be used in conjunction with the variational Bayes clustering to allow much more information to be incorporated into the plot and allow substructure to be visualised. It was also shown how other information regarding the genes could be easily incorporated into this type of visualisation, to observe the spread of certain genes through the data space, both across the different clusters and within the different clusters. Even though chapter 4 suggested the presence of angular structures in the data, and that projecting the data into an angular space, defined by a series of pairwise comparisons, may be advantageous for the data sets considered in this thesis, these may not be present or of interest in all data sets. For some studies, it may be of interest to observe the gene groupings based on their log fold changes rather than relative changes in expression. As essentially the same methodology can be used for the visualisation of both the log fold data and the angular data then both were demonstrated in chapter 6. Given the computational efficiency of the methods, there is no reason why different data projections or series of data projections should not be carried out on a data set and the groupings in the different visualisations compared as the different data projections highlight different aspects of the data. In order to get most information from the data set and the different projection methods it is however important to understand the effect the transformations will have on the data and exactly how the different methods have modelled patterns within the data.

8.1.3 Combinations of Transcription Factor Binding Site Motifs

Gene expression patterns hidden inside gene expression data sets can be found but what is of more interest is why the genes show the same expression patterns. Chapter 2 discussed how the expression of genes is controlled by the binding of transcriptions factors to DNA motifs. Therefore, genes with similar expression profiles may be controlled by similar profiles and contain similar binding motifs in the DNA sequence close to the gene. Previous models for locating transcription factor binding sites have concentrated on finding
individual sites in individual sequences and have a high false positive rate. As multiple transcription factors interact to affect gene regulation, attempts to improve the false positive rate by looking for clusters of motifs, as clusters will have a much lower false positive rate. These models can be effective if there is prior knowledge such as the expected number of each motif and the gaps between them, but this is not always available. Other models search through different possible combinations of motifs. Chapter 7 introduced a method which can be used to search combinations of binding sites by looking for them in the upstream/downstream regions of multiple genes. Thus, the false positive rate is reduced by finding combinations that occur multiple times across co-expressed genes. The suggestions for co-expressed groups of genes can be obtained directly from the results of the variational Bayes clustering or angular distribution decomposition (circular variational Bayes clustering of angular transformed data). The method uses non-negative matrix factorisation to decompose a motif score matrix into two other matrices, and therefore can be thought of as another data projection method, in which the information from the motif score matrix is projected into two separate data spaces. This allows the genes to be described in terms of a set of underlying patterns which correspond to the combinations of motifs. Rather than requiring a continuous sequence, multiple DNA sequences can be examined, such as sequences much further away from the start site that could contain enhancers or the reverse or complementary sequences to account for when a transcription factor might bind in the opposite direction or on the other strand of DNA. It is also possible to use the same technique to look for combinations of protein domain sequences in groups of proteins to look for patterns of functional sites.

8.1.4 Circular Data Analysis

In chapter 5 the variational Bayes mixture modelling approach was adapted for use with wrapped Normal distributions. Similarly, chapter 6 showed how the GTM model could be adapted for use with circular data. These techniques could be applied to the analysis and visualisation of any angular data set and are not restricted for use gene expression data.
8.2 Further Work

8.2.1 Visualisation

Relevance Dimensions

The idea of relevance dimensions discussed in Yip et al. (2005), are lower dimensional clusters in higher dimensional space. The relevance dimensions are the dimensions along which data is significantly close to each other and the different clusters in the data may have different relevance dimensions. An example of this was seen in the yeast analysis, where cluster A seemed to be tightly clustered along all dimensions (pairwise comparisons), but cluster B was only clustered in the comparisons relating to the heat shock treatments (figure 8.1).

It can be seen that comparisons 1-7 are relevant for both clusters A and B, and that they two clusters have similar means, thus cluster A is a group of data points contained within the data for cluster B. The mixture modelling approach was able to distinguish these groups despite the difference in relevance dimensions and the overlapping of the clusters, by allowing the covariance parameters to be much larger for those dimensions that are not relevant in cluster B. However, in the GTM projection in 2D the two clusters are overlapping (figure 8.2), with cluster B spread over much more of the data space than cluster A. This is because of the much larger variation in the non-relevant dimensions, which will mean different parts of the manifold will be used to model different parts of the cluster.
Figure 8.2: Angular GTM visualisation of the yeast data with clusters A and B highlighted. The two clusters are overlapping but Cluster B is spread over greater area of manifold than cluster A. This is because of the variation in the data for cluster B along the dimensions that are not relevance dimensions.

The information in the covariance matrix could be used to make a choice about the relevance dimensions for a particular cluster, and therefore project each cluster into a lower dimensional space that is most appropriate. Although it is not possible to use the GTM to create a visualisation with the clusters projected from different dimensions, when looking for substructure within a particular cluster by using the GTM to project only the data points within a single cluster, then a better visualisation may be given by using only the relevance dimensions.

**Gaussian Process Methods**

Recently Gaussian Processes have become a popular tool for machine learning problems (Rasmussen and Williams, 2006). They work by defining a probabilistic distribution over a function rather than data samples. A function can be thought of as a mapping from one data space to another and for any data set there might be multiple functions that agree with the data samples in describing the mapping between the two data spaces, thus there is an uncertainty as to which function is the correct model. In regions of the data spaces where the mapping is known (by the data samples in the data set) then there is very low uncertainty, however in regions where there are no examples in the data set then there is larger uncertainty in the function that maps between the two data spaces. A Gaussian process describes the distribution and uncertainty over the different mappings across the entire data spaces using the concept of a mean function and covariance function.
Lawrence (2005) showed how Gaussian Process Latent Variable Models (GPLVMs) could be used for dimensionality reduction problems. Using a covariance kernel corresponding to a radial basis function mapping between the latent space and data space, the model is similar to the GTM algorithm. In order to learn the mapping between the two spaces marginalisation over both the latent variables and weight matrix for the RBF network is required, however this is intractable, so in practice the model is marginalised over one of them and then optimal values are found for the others by maximising the likelihood. GTM marginalises over the latent variables by defining them as a fixed grid of delta functions, and then optimises the weight matrix of the RBF network. In contrast, GPLVM marginalises over the weight matrix and then optimises the value of a latent variables for each of the data points (the positions of the data points in latent space). This optimisation over latent points means that there is no artificial ‘discretisation’ in the visualisations using GPLVMs which can appear in GTM because of the fixed grid of latent variables. Fixing the latent variables across the latent space also has the effect of causing the manifold to be stretched and compressed in the data space, as all areas of the manifold are used to fit data points. This therefore also causes artificial stretching of areas of high data density and compressing of low areas of data density when mapped into 2D, making it difficult to see discrete clusters, even with the aid of magnification factors. The GPLVMs importantly provide information about the certainty of the mapping from each co-ordinate in latent space to co-ordinates in the data space. This uncertainty can be used to visualise groups of data points, as the uncertainty will be small in areas that map to regions of the data space where there are data points and high in areas where there are no data points. Therefore, if the data is clustered in the data space, the visualisation will show data points in areas of low uncertainty separated by areas of high uncertainty. The overall effect of using Gaussian Processes to optimise the latent variables can therefore lead to much smoother visualisations.

A possible extension is to use GPLVMs to visualise the gene expression data and to investigate the effects of using different covariance kernels to model the mapping between the
8.2 Further Work

8.2.2 Motif Analysis

Generating Position Weight Matrices

Currently a major problem for locating binding sites is the quality and small number of position weight matrices (PWMs) that are available from experimental data. Thus algorithms that predict PWMs could be used to provide many different hypotheses for individual sites, and the NNMF method used to look for combinations of these sites, investigate which are similar or if any are not associated with the genes and thus bad predictions.

There are many algorithms that have been designed to predict binding sites, in which they model the sites as short sequences that are over-represented within a set of much longer sequences that correspond to the promoter regions of a set of co-regulated genes. These techniques include strategies based on Gibbs sampling (Lawrence et al., 1993; Roth et al., 1998; Thijs et al., 2001; Liu et al., 2001); expectation maximisation (Lawrence and Reilly, 1990; Bailey and Elkan, 1994); word enumeration approaches (van Helden et al., 1998; Vilo et al., 2000; Liu et al., 2002; Bussemaker et al., 2000); tree search approaches (Pavesi et al., 2001, 2004; Eskin and Pevzner, 2002); z-scores (Sinha and Tompa, 2000, 2002, 2003) and variational Bayes methods (Xing et al., 2003). However, these methods require different parameterisations and give very different predictions. They can also be sensitive to the set of sequences, length of sequences and width of motifs searched for, so even the same algorithm may provide very different predictions when used with slightly different sequences or parameters. As it is not known exactly which sequences will share binding sites and exactly how far upstream/downstream of the gene they motifs can be found, then it is difficult to choose a set of promoter sequences that will provide good predictions.

Therefore, one possible way to generate PWMs is to predict from a mixture of algorithms with multiple sets of sequences, different motif widths and different sequence windows lengths upstream of the transcription start site. It would be difficult to find any patterns from all these different predictions, but the construction of a motif score matrix and NNMF technique should be able to deal with all these different suggestions, even if they are of poor
quality or do not really correspond to motifs found in the region of multiple sequences. PWMs that are very similar models will provide similar motif scores and appear as combinations in the matrix decomposition, motif suggestions that do not appear very frequently in the sequences will be treated as noise and motifs that appear very frequently, in most sequences will also be modelled as a combination that appears in all sequences. Therefore, having too many predictions may be far better than to have too few. In order to highlight the patterns an iterative approach could be taken to combine or remove motif models that are the same by analysing the similarities of the models in each combination found. Models that appear very infrequently or appear in all sequences could also be removed, so that the matrix factorisation concentrates on finding the patterns of different motifs that could explain the differences in regulation between different gene groups.

A second source of motif model predictions can be found by searching the sequences for patterns that are associated with certain families of transcription factors. For instance, many transcription factors are proteins built from multiple subunits, with each subunit having a DNA binding domain and attaching to a different section of the DNA sequence in the promoter region. Figure 8.3 shows examples of zinc fingers, helix-turn-helix and leucine zipper proteins that are constructed from multiple subunits. Although the zinc finger shown only has two subunits, zinc fingers can contain more than two subunits. Some transcription factors are dimers, having two of the same subunits, so the two binding domains will be the same and the motif model for this will contain two repeated sequences with a gap in between the repeats. If the subunits are configured such that they bind opposite strands in opposite directions, then the motifs will contain a sequence which binds the first subunit, followed by a gap, followed by the reverse complement of the sequence (termed palindromic repeats). Thus, searching for sub-sequences that contain direct and palindromic repeats may provide suggestions for motif models, as well as analysing sub-sequences that contain multiple sequences associated with binding the subunits of zinc finger proteins.
8.2 Further Work

Figure 8.3: Some families of transcription factors: zinc finger, helix-turn-helix, leucine zipper. These types of transcription factors are made from multiple subunits, each subunit has a binding domain and binds the DNA at particular short sequences. Thus the overall motif model of the transcription will be the short sequences that bind the different subunits with gaps in between that could be any sequence of bases. If the transcription factor is a dimer, made from two subunits that are the same, then the binding domains for the subunit will be the same, so the motif will contain a direct repeat or palindromic repeat if the subunits bind different strands. Image reproduced from http://www.uic.edu/classes/bios/bios100/lecturesf04am/lect15.htm.

Model Order Selection

In the examples shown in chapter 6 the model order (number of combinations) was selected manually. However, it would be better to have an automated method for model order selection. The NNMF implementation works by finding a local minimum for the sum squared error between the elements in the original motif score matrix and the approximation to them in $UV$. Having too few combinations means that not all the motif combinations are extracted, or in some cases it can be a very poor approximation that over generalises the importance of a combination, and the motifs appear in the approximation where they did not appear in the original score matrix. The greater the number of combinations allowed, the smaller the error in the approximation will be, but if motifs that do not appear frequently or in combinations are to be treated as noise, then these need to be removed by the approximation. Therefore, a balance is required between having sufficient combinations to model the motif patterns and having the least complex model of motif patterns. Methods that allow this model order selection should be investigated.

Results extraction

The examples in chapter 6 were demonstrated by visualising the matrices as images. This can be a useful way to visualise any structure or patterns in the matrices. However, if the technique is used for a much larger number of sequences or motifs, then any common
Further Work

patterns would not be so easy to see. Therefore, the visualisation may need to be improved by clustering the sequences according to their motif scores in the motif score matrix, or clustering with regard to the scores for each motif combinations for $V$. Also, it would be useful to output a list of motifs associated with each combination and the sequences associated with each combination, so that the information can be used to direct researchers to particular genes. To do this, threshold values would be required, above which the motif or sequence is considered associated with a combination. The best way to choose these values needs to be investigated.

8.2.3 Biological Research

Software

This thesis has discussed different data projections that can be applied to bioinformatics data in order to highlight different patterns and/or groupings of points within the data sets. However, for the analysis and visualisation to be useful to biological researchers, then a software tool is really required that incorporates the different data projections and allows the different stages of analysis to be done. It would also be useful to incorporate further sources of information, linking to databases and other analysis techniques such as BLAST analysis, which compares gene and protein sequences with sequences found in other organisms.
Part I

Appendix
Circular EM and Circular VB Mixture Modelling

This appendix provides details of how to implement the EM and Variational Bayes algorithms for learning the parameters of a mixture of $K$ multi-dimensional wrapped Normals to model some angular data $A$. $A = \{a_{ij}\}$, with $a_{ij} \in [0, 2\pi]$ for $i = 1...N$ and $j = 1..C$, $C$ is the dimensionality of the angular data. The $k^{th}$ wrapped normal component has an associated mixing probability $\pi_k$, mean $\mu_k$ and precision $\beta_k$. $L_n$ is the mixture label of the $n^{th}$ datum. The variable $z^\mu_k_n$ is defined in equation 5.10 and gives the wrapped version of angular data for the $n^{th}$ data point that is closest to $\mu_k$.

A.1 Circular Expectation-Maximisation

This uses the same methodology as given in section 3.3.5 but with the equations modified to ensure the minimum wrapped distances between the data and component means are correctly calculated. The log likelihood of the entire data set can be computed by summing the log likelihood for all datapoints:

$$\mathcal{L} = \sum_{n=1}^{N} \log p(a_n)$$  \hspace{1cm} (A.1)

where

$$p(a_n) = \sum_{k=1}^{K} \pi_k WN(a_n; \mu_k, \beta_k)$$  \hspace{1cm} (A.2)

**E-step:** Infer the posterior probabilities, $p(k|a_n)$, of the mixture labels given each datum. For each wrapped normal component $k = 1...K$ we have

$$p(k|a_n) = p(L_n = k|a_n) = \frac{\pi_k WN(a_n; \mu_k, \beta_k)}{p(a_n)}$$  \hspace{1cm} (A.3)
M-step: re-estimate the parameters for each component in the mixture model given the updated E-step posteriors. The following set of update equations are used:

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} p(k|a_n), \quad \mu_k = \frac{\sum_{n=1}^{N} z_n^{(k)} p(k|a_n)}{\sum_{n=1}^{N} p(k|a_n)}$$

and

$$\beta_k^{-1} = \frac{\sum_{n=1}^{N} (z_n^{(k)} - \mu_k)^T (z_n^{(k)} - \mu_k) p(k|a_n)}{\sum_{n=1}^{N} p(k|a_n)}$$ (A.4)

The number of components to use can be found using the BIC score given in equation 3.16, which compares models with different numbers of components to find the optimal trade off between log likelihood and model complexity.

### A.2 Circular Variational Bayes

The log evidence is given by equation A.5, and as it is constant with respect to $\Theta$, minimising the KL-divergence between the true and approximating posteriors is equivalent to maximising the negative free energy $F[q(\Theta|A)]$. Model complexity can be automatically penalised in the variational Bayes approach, with components removed if they are not required, so only one model needs to be calculated, provided it has a sufficient number of components. Thus $K$ is an upper limit on the number of components in the mixture.

$$\log p(A) = \int_q q(\Theta|A) \log \frac{p(A, \Theta)}{q(\Theta|A)} d\Theta + \int_{\Theta} q(\Theta|A) \log \frac{q(\Theta|A)}{p(\Theta|A)} d\Theta$$

$$= F[q(\Theta|A)] + KL[q(\Theta|A)||p(\Theta|A)]$$ (A.5)

### A.2.1 Priors

The prior on the model parameters is separable such that:

$$p(\Theta) = p(\pi) \prod_k p(\mu_k)p(\beta_k)$$ (A.6)
in which
\[ p(\pi) = \mathcal{D}(\lambda_0 I_K) \]
\[ p(\beta_k) = \mathcal{W}(\alpha_0, B_0) \]  \hspace{1cm} (A.7)
\[ p(\mu_k|\beta_k) = \mathcal{N}(m_0, \tau_0 \beta_k) \]
where \( I \) is the identity matrix. \( \mathcal{D}, \mathcal{W}, \mathcal{N} \) indicate the Dirichlet, Wishart and Normal distributions respectively. The prior for the means, \( p(\mu_k|\beta_k) \), is conditioned on the precisions. The joint prior \( p(\mu_k, \beta_k) \) would be thus be a Normal-Wishart distribution.

The priors were set as follows:
\[ \lambda_0 = \{\lambda_0^1, \ldots, \lambda_0^K\}, \quad \lambda_0^i = 1 \quad \text{for} \quad i = 1..K \]
\[ \alpha_0 = C + 1 \]
\[ B_0 = (A_{\text{max}} - A_{\text{min}}) I_C \]
\[ m_0 = E[A] \]
\[ \tau_0 = 1 \]  \hspace{1cm} (A.8)

A.2.2 Posterior

The approximating posterior is separable such that:
\[ q(\Theta|A) = q(\pi|A) \prod_k q(\mu_k|A)q(\beta_k|A) \]  \hspace{1cm} (A.9)
where
\[ q(\beta_k|A) = \mathcal{W}(\alpha_k, B_k) \]
\[ q(\mu_k|\beta_k, A) = \mathcal{N}(m_k, \tau_k \beta_k) \]
\[ q(\pi|A) = \mathcal{D}(\lambda) \]  \hspace{1cm} (A.10)

A.2.3 Update Equations

E-Step:

The E-step consists of updating the label posteriors \( q(L_n = k|A, \Theta) \).
\[ \tilde{q}(L_n = k | A, \Theta) = \tilde{\pi}_k \tilde{\beta}_k^T \exp \left( -\frac{1}{2} (z_{mk}^n - m_k)^T \tilde{\beta}_k (z_{mk}^n - m_k) \right) \exp \left( -\frac{C}{2\tau_k} \right) \]  

(A.11)

where

\[ \log \tilde{\pi}_k = \Psi(\lambda_k) - \Psi \left( \sum_{k'} \lambda_{k'} \right) \]

\[ \log \tilde{\beta}_k = \sum_{i=1}^C \Psi((\alpha_k + 1 - i)/2) - \log |B_k| + C \log 2 \]

\[ \tilde{\beta}_k = \alpha_k B_k^{-1} \]

\[ \Psi() \] is the digamma function.

The label posteriors are normalised to give:

\[ q(L_n = k | A, \Theta) = \frac{\tilde{q}(L_n = k | A, \Theta)}{\sum_{k'} \tilde{q}(L_n = k' | A, \Theta)} \]  

(A.13)

**M-Step:**

The model hyperparameters are updated as follows:

\[ \lambda_k = \bar{N}_k + \lambda_0 \]

\[ m_k = \frac{\bar{N}_k \bar{\mu}_k + \tau_0 m_0}{\bar{N}_k + \tau_0} \]

\[ \tau_k = \bar{N}_k + \tau_0 \]  

(A.14)

\[ \alpha_k = \bar{N}_k + \alpha_0 \]

\[ B_k = \bar{N}_k \bar{\Sigma}_k + \frac{\bar{N}_k \tau_0 (\bar{\mu} - m_0)^T (\bar{\mu} - m_0)}{\bar{N}_k + \tau_0} + B_0 \]

where

\[ \bar{\pi}_k = \frac{1}{N} \sum_{n=1}^N q(L_n = k | A, \Theta) \]

\[ \bar{N}_k = N \bar{\pi}_k \]

\[ \bar{\mu}_k = \frac{1}{N_k} \sum_{n=1}^N q(L_n = k | A, \Theta) z_{mk}^n \]

\[ \bar{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^N q(L_n = k | A, \Theta) (z_{mk}^n - \bar{\mu}_k)^T (z_{mk}^n - \bar{\mu}_k) \]  

(A.15)
A.3 Initialisation of Model Parameters

It can be seen that the hyperparameters are updated based on two terms, one gives the value of the hyperparameter according to the prior estimates, the second gives an estimate of the hyperparameter based on the data and is weighted by the proportion of data that the component is modelling. Thus, as a component has a lower probability of modelling data points, the estimates converge back to the priors.

A.3 Initialisation of Model Parameters

The circular EM algorithm requires initial estimates for the parameters and the circular VB algorithm requires initial estimates for the posterior hyperparameters. Hierarchical clustering can be used to find these initial estimates, in which $N$ clusters are initialised, each containing a single data point, then the closest clusters, according to the wrapped distance measure, joined using average linkage until only $K$ clusters remain. The wrapped mean for cluster $k$ can be found by aligning each of the data points, $a_n$ in cluster $k$ with a reference sample $a_r$, for example the first element in the cluster $k$ list. $z_{n}^{a_r}$ is the version of $a_n$ with the smallest wrapped distance to $a_r$. $A^k$ is the set of data points allocated to cluster $k$ and $N_k$ is the number of points allocated to cluster $k$.

A.3.1 Circular EM

\[
\pi_k^{\text{init}} = \frac{N_k}{N} \\
\mu_k^{\text{init}} = \left( \frac{1}{N_k} \sum_{a_n \in D_k} z_{n}^{a_r} \right) \mod 2\pi \\
\beta_k^{\text{init}} = \left( \frac{1}{N_k} \sum_{a_n \in A^k} (z_{n}^{a_r} - \mu_k^{\text{init}})^T (z_{n}^{a_r} - \mu_k^{\text{init}}) \right)^{-1}
\] (A.16)
Figure A.1: Analysis of the Rand indices from the clustering results compared with the known cluster labels for the data sets (a) Histogram of the highest Rand index obtained for each data set (b) Histogram of differences in Rand index between the result from circular VB initialised with 25 clusters and the circular EM result suggested as optimal by BIC. The histogram is skewed showing that the Bayesian approach of approximating the posterior distribution gives better results than penalised maximum likelihood.

A.3.2 Circular VB

\begin{equation}
\begin{align*}
\lambda_k^{\text{init}} &= \frac{N_k}{N}, \\
m_k^{\text{init}} &= \left[ \frac{1}{N_k} \sum_{a_n \in D_k} z_{a_n}^{a_r} \right] \mod 2\pi \\
\tau_k^{\text{init}} &= 1; \\
\alpha_k^{\text{init}} &= 2\alpha_0; \\
\beta_k^{\text{init}} &= \alpha_k^{\text{init}} \left[ \frac{1}{N_k} \sum_{a_n \in A_k} (z_{a_n}^{a_r} - \mu_k^{\text{init}})^T (z_{a_n}^{a_r} - \mu_k^{\text{init}}) \right].
\end{align*}
\end{equation}

(A.17)

A.4 Comparison of Circular EM and Circular VB

Simulated data sets were generated from mixtures of multivariate normals with different covariance matrices and means, the means being in the range [0..2\pi]. The dimensions of the data sets ranged from 5 to 20 and the number of clusters also ranged from 5 to 20. Half of the data sets were created with equal cluster sizes, and half had cluster sizes randomly chosen.

In total 96 data sets were created, and for each, both the circular EM and circular VB algorithms were run with initialisations of 2 to 25 clusters. The quality of the clustering results were assessed using the Rand index to compare the partitionings given by the algorithms with the clusters that the data was generated from. The Rand index for two
clustering results is a value between 0 and 1, a value of 1 means that the results are the same and hence the algorithm has perfectly clustered the data.

Figure A.1(a) shows the highest Rand indices obtained for 81 of the 96 data sets. The EM algorithm failed to successfully converge with all initialisations on the remaining 15 data sets due to underflow errors, so we compare the methods on the 81 data sets that were not affected. Both algorithms produce good models of the simulated data, the smallest Rand index for the EM algorithm being 0.9597 and 0.9603 for VB with the majority being greater than 0.995.

Figure A.1(b) shows the differences in Rand index between the VB model initialised with 25 clusters and the optimal EM model according to the BIC criteria. The histogram is heavily skewed, showing that the VB approach gives the same or better Rand index than using BIC. In summary, the VB approach is not only faster but consistently gives good results on the simulated data, provided that it is initialised with a sufficient number of clusters.
Pre-processing of the Rice Blast Data set

B.1 Description of Microarrays

Ten cDNA microarrays were used to compare the gene expression in the *Magnaporthe grisea* organism under five different conditions. The microarrays showed the expression of 20810 genes, 13666 of which were *Magnaporthe grisea* genes and the remaining 7144 were rice plant genes. The five experimental conditions were as follows:

1. Spore
2. 7 hours after spore on a hydrophobic surface
3. 12 hours after spore on a hydrophobic surface
4. 7 hours after spore on a hydrophilic surface
5. 12 hours after spore on a hydrophilic surface

Different combinations of the five conditions were hybridised to each microarray slide. The arrows in figure B.1 show which conditions were hybridised to each slide. The arrows point in the direction of the red channel, for example slide 531 hybridised the hydrophobic 12 hours sample in the red channel and the spore sample in the green channel.

B.2 Pre-processing

The mean signal intensity values and the mean background intensities for each spot were extracted from the files. The background values were then subtracted from the mean signal intensities to give a red and a green value for the expression of each gene on each slide.
B.3 Correcting for Dye Biases

B.3.1 Distribution of Data in Each Channel

Box plots were made of the different channels from each slide (see figure B.2). The boxplots show that the distribution of the data varied between the channels with the red dye having a more variable expression change. If there was no dye bias then it was expected that the mean intensity value would be similar across all channels as the majority of genes should show similar expression levels in all channels. However the mean clearly differed.
B.3 Correcting for Dye Biases

Figure B.2: Box plots showing the distribution of the data in each of the channels, indicating variation introduced by the dyes.

B.3.2 Distribution of Log Ratios on Each Slide

Similarly it was expected that the log ratio intensities on each slide would be centred at zero (see figure B.3c). However the majority of slides showed bias towards the green dye as they had a negative average intensity. As the green dye was used with different experimental conditions in each slide then the bias was likely to be due to the dye and not a particular condition.

B.3.3 Lowess Regression

In order to remove the dye biases an MA plot was made and then loess regression performed in the manner suggested by Yang et al. (2002). A MA-plot plots the average intensity for each gene against the log ratio intensity. An example of a MA plot for slide S531 is shown in figure B.3a. The MA plots for all the slides showed a similar shape. The MA-plot confirmed a bias towards the green dye particularly at low average intensities. Figure B.3a also shows the loess regression line which fits the centre of the data. Figure B.3b shows the MA plot after the log ratios have been normalised.

B.3.4 Distribution of Log Ratios After Normalisation

The distribution of the normalised log ratios are shown in figure B.3d, which shows a clear improvement to the distributions before normalisation. The distributions for each slide are
B.3 Correcting for Dye Biases

a) MA plot of slide S531  

b) MA plot after normalisation

c) Spread of Data on each slide  
d) Spread of Data after normalisation

Figure B.3: MA plot of slide S531 (a) before and (b) after adjustment using loess regression. Distribution of data on the slides (c) before and (d) after normalisation.

centred close to zero and all slides have similar spreads in intensity.

The slides could then be mean (or median) centred to make them comparable, but it was felt that as the spreads were very similar, scale adjustment was not necessary and could actually introduce errors.
Figure B.4: MVA plots before and after lowess regression has been applied for slides 531, 532, 533, 534 and 535.
B.3 Correcting for Dye Biases

Figure B.5: MVA plots before and after lowess regression has been applied for slides 541, 542, 543, 544 and 545.
Angular Distribution Decomposition
Cluster Information - Rice Blast Data

The cluster information for the data set of the rice blast *Magnaporthe grisea* during infection-related development. The conditions of study in this data set represent: ungerminated spores (time 0), tissues harvested from hydrophobic surface, (I), and hydrophilic surface, (NI) at 7 hours ( appressorium forming stage only on I) and 12 hours post inoculation (appressorium and penetration peg only on I), as detailed in Dean *et al.*, 2005.

The four arrays comparing treatments after germination with the ungerminated spores were selected for study. 2968 genes were then chosen for further study as those that showed a 2-fold or larger expression change in at least one condition. Pairwise comparisons of the same surface but different time point and same time point but different surfaces were made.

C.1 Conditions of study
1. I 7h /sp
2. I 12h /sp
3. NI 7h /sp
4. NI 12h /sp

C.2 Pairwise Comparisons
1. I 7h /sp with I 12h /sp
2. NI 7h /sp with NI 12h /sp
3. I 7h /sp with NI 7h /sp
4. I 12h /sp with NI 12h /sp
C.3 Clusters

C.3.1 Cluster A

Cluster size: 129

Component parameters

<table>
<thead>
<tr>
<th></th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>1.0326</td>
<td>0.23881</td>
<td>1.1781</td>
</tr>
<tr>
<td>Angle 2</td>
<td>2.7802</td>
<td>1.6984</td>
<td>2.7489</td>
</tr>
<tr>
<td>Angle 3</td>
<td>6.1439</td>
<td>0.17057</td>
<td>0</td>
</tr>
<tr>
<td>Angle 4</td>
<td>0.038273</td>
<td>0.17853</td>
<td>0</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. More up-regulation in I 12h /sp than I 7h /sp
2. Down-regulation in NI 7h /sp. Relatively small up-regulation in NI 12h /sp
3. Up-regulation in I 7h /sp. No change in NI 7h /sp
4. Up-regulation in I 12h /sp. No change in NI 12h /sp

Angular data

Log fold data
C.3 Clusters

C.3.2 Cluster B

Cluster size: 98

Component parameters

<table>
<thead>
<tr>
<th></th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>1.6403</td>
<td>0.20529</td>
<td>1.5708</td>
</tr>
<tr>
<td>Angle 2</td>
<td>3.4124</td>
<td>0.60669</td>
<td>3.5343</td>
</tr>
<tr>
<td>Angle 3</td>
<td>4.4131</td>
<td>0.50245</td>
<td>4.3197</td>
</tr>
<tr>
<td>Angle 4</td>
<td>6.0841</td>
<td>0.32396</td>
<td>5.8905</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. No change in I 7h /sp. Up-regulation in I 12h /sp
2. More down-regulation in NI 7h /sp than NI 12h /sp
3. More down-regulation in NI 7h /sp than I 7h /sp
4. Up-regulation in I 12h /sp. Relatively small down-regulation in NI 12h /sp

Angular data

![Angular data graph]

Log fold data

![Log fold data graph]
C.3.3 Cluster C

Cluster size: 1150

Component parameters

<table>
<thead>
<tr>
<th>Angle</th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>0.7086</td>
<td>0.2013</td>
<td>0.7854</td>
</tr>
<tr>
<td>Angle 2</td>
<td>0.70113</td>
<td>0.053712</td>
<td>0.7854</td>
</tr>
<tr>
<td>Angle 3</td>
<td>0.90268</td>
<td>0.12158</td>
<td>0.7854</td>
</tr>
<tr>
<td>Angle 4</td>
<td>0.92429</td>
<td>0.17352</td>
<td>0.7854</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. Similar up-regulation in I 7h/sp and I 12h/sp
2. Similar up-regulation in NI 7h/sp and NI 12h/sp
3. Similar up-regulation in I 7h/sp and NI 7h/sp
4. Similar up-regulation in I 12h/sp and NI 12h/sp

Angular data

![Angular data plot](image)

Log fold data

![Log fold data plots](image)
C.3 Clusters

C.3.4 Cluster D

Cluster size: 1027

**Component parameters**

<table>
<thead>
<tr>
<th>Angle</th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>3.9527</td>
<td>0.044228</td>
<td>3.927</td>
</tr>
<tr>
<td>Angle 2</td>
<td>3.86</td>
<td>0.040895</td>
<td>3.927</td>
</tr>
<tr>
<td>Angle 3</td>
<td>3.9343</td>
<td>0.053935</td>
<td>3.927</td>
</tr>
<tr>
<td>Angle 4</td>
<td>3.8477</td>
<td>0.062541</td>
<td>3.927</td>
</tr>
</tbody>
</table>

**Interpretation of mean angles**

1. Similar down-regulation in I 7h /sp and I 12h /sp
2. Similar down-regulation in NI 7h /sp and NI 12h /sp
3. Similar down-regulation in I 7h /sp and NI 7h /sp
4. Similar down-regulation in I 12h /sp and NI 12h /sp

**Angular data**

![Angular data graph](image)

**Log fold data**

![Log fold data graphs](image)
C.3 Clusters

C.3.5 Cluster E

Cluster size: 167

Component parameters

<table>
<thead>
<tr>
<th>Angle</th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>5.4644</td>
<td>0.98985</td>
<td>5.4978</td>
</tr>
<tr>
<td>Angle 2</td>
<td>4.2284</td>
<td>0.85182</td>
<td>4.3197</td>
</tr>
<tr>
<td>Angle 3</td>
<td>5.5884</td>
<td>1.4874</td>
<td>5.4978</td>
</tr>
<tr>
<td>Angle 4</td>
<td>4.1259</td>
<td>0.60695</td>
<td>4.3197</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. Up-regulation in I 7h /sp. Down-regulation in I 12h /sp
2. More down-regulation in NI 12h /sp than NI 7h /sp
3. Up-regulation in I 7h /sp. Down-regulation in NI 7h /sp
4. More down-regulation in NI 12h /sp than I 12h /sp

Angular data

Log fold data
C.3.6 Cluster F

Cluster size: 135

Component parameters

<table>
<thead>
<tr>
<th></th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>5.3528</td>
<td>0.55189</td>
<td>5.4978</td>
</tr>
<tr>
<td>Angle 2</td>
<td>6.2279</td>
<td>0.83237</td>
<td>0</td>
</tr>
<tr>
<td>Angle 3</td>
<td>0.98493</td>
<td>0.55555</td>
<td>1.1781</td>
</tr>
<tr>
<td>Angle 4</td>
<td>2.9713</td>
<td>0.8878</td>
<td>3.1416</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. Up-regulation in I 7h /sp. Down-regulation in I 12h /sp
2. Up-regulation in NI 7h /sp. No change in NI 12h /sp
3. More up-regulation in NI 7h /sp than I 7h /sp
4. Down-regulation in I 12h /sp. No change in NI 12h /sp

Angular data

Log fold data
C.3 Clusters

C.3.7 Cluster G
Cluster size: 68

Component parameters

<table>
<thead>
<tr>
<th>Angle</th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>1.102</td>
<td>0.37096</td>
<td>1.1781</td>
</tr>
<tr>
<td>Angle 2</td>
<td>5.3999</td>
<td>0.44871</td>
<td>5.4978</td>
</tr>
<tr>
<td>Angle 3</td>
<td>0.81134</td>
<td>0.63888</td>
<td>0.7854</td>
</tr>
<tr>
<td>Angle 4</td>
<td>5.7471</td>
<td>0.43772</td>
<td>5.8905</td>
</tr>
</tbody>
</table>

Interpretation of mean angles
1. More up-regulation in I 12h /sp than I 7h /sp
2. Up-regulation in NI 7h /sp. Down-regulation in NI 12h /sp
3. Similar up-regulation in I 7h /sp and NI 7h /sp
4. Up-regulation in I 12h /sp. Relatively small down-regulation in NI 12h /sp

Angular data

Log fold data
C.3 Clusters

C.3.8 Cluster H

Cluster size: 140

Component parameters

<table>
<thead>
<tr>
<th></th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>3.521</td>
<td>0.67591</td>
<td>3.5343</td>
</tr>
<tr>
<td>Angle 2</td>
<td>4.5263</td>
<td>0.20695</td>
<td>4.7124</td>
</tr>
<tr>
<td>Angle 3</td>
<td>3.6435</td>
<td>0.37353</td>
<td>3.5343</td>
</tr>
<tr>
<td>Angle 4</td>
<td>4.4355</td>
<td>0.38004</td>
<td>4.3197</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. More down-regulation in I 7h /sp than I 12h /sp
2. No change in NI 7h /sp. Down regulation in NI 12h /sp
3. More down-regulation in I 7h /sp than NI 7h /sp
4. More down-regulation in NI 12h /sp than I 12h /sp

Angular data

Log fold data
C.3 Clusters

C.3.9 Cluster I

Cluster size: 54

Component parameters

<table>
<thead>
<tr>
<th>Angle</th>
<th>Expected value of mean</th>
<th>Expected variance</th>
<th>Most probable reference angle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle 1</td>
<td>2.1824</td>
<td>0.53806</td>
<td>2.3562</td>
</tr>
<tr>
<td>Angle 2</td>
<td>1.0591</td>
<td>0.36319</td>
<td>1.1781</td>
</tr>
<tr>
<td>Angle 3</td>
<td>2.2219</td>
<td>0.5461</td>
<td>2.3562</td>
</tr>
<tr>
<td>Angle 4</td>
<td>0.95582</td>
<td>0.46895</td>
<td>0.7854</td>
</tr>
</tbody>
</table>

Interpretation of mean angles

1. Down-regulation in I 7h /sp. Up-regulation in I 12h /sp
2. More up-regulation in NI 12h /sp than NI 7h /sp
3. Down-regulation in I 7h /sp. Up-regulation in NI 7h /sp
4. Similar up-regulation in I 12h /sp and NI 12h /sp

Angular data

Log fold data
Position Weight Matrices (PWM) Examples

This appendix gives the position weight matrices predicted from MEME in the analysis of the upstream promoter regions described in section 7.4.1.

<table>
<thead>
<tr>
<th>Cluster A - Motif Prediction 1</th>
<th>Consensus: TTTTTTTTTCTTTT</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 0.00 0.00 0.06 0.00</td>
<td>0.07 0.10 0.00 0.01 0.01 0.01</td>
</tr>
<tr>
<td>C 0.17 0.40 0.00 0.01 0.04 0.17 0.03 0.00 0.11 0.17</td>
<td>0.53 0.00 0.15 0.25 0.17</td>
</tr>
<tr>
<td>G 0.00 0.00 0.00 0.17 0.14 0.00 0.06 0.09 0.01 0.03</td>
<td>0.00 0.00 0.04 0.00 0.00</td>
</tr>
<tr>
<td>T 0.83 0.60 0.94 0.82 0.75 0.74 0.92 0.90 0.86 0.79</td>
<td>0.26 0.96 0.81 0.75 0.78</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cluster A - Motif Prediction 2</th>
<th>Consensus: TTTTTTTTTCTC</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 0.00 0.04 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
<td>0.00 0.00</td>
</tr>
<tr>
<td>C 0.40 0.00 0.30 0.37 0.37 0.00 0.13 0.07 0.00 0.63</td>
<td>0.00 0.59</td>
</tr>
<tr>
<td>G 0.00 0.00 0.00 0.00 0.00 0.02 0.00 0.22 0.00 0.11</td>
<td>0.00 0.00</td>
</tr>
<tr>
<td>T 0.61 0.96 0.70 0.63 0.63 0.98 0.87 0.72 1.00 0.26</td>
<td>1.00 0.41</td>
</tr>
</tbody>
</table>

Table D.1: The T-rich motifs in combination 1 of the NNMF analysis for the promoter regions of cluster A (figure 7.9).
### Table D.2: The T-rich motifs/A-rich motif in combination 1 of the NNMF analysis for the promoter regions of cluster B (figure 7.10).

<table>
<thead>
<tr>
<th>Cluster B - Motif Prediction 1</th>
<th>Consensus: TTTTTTTTTCTTTTT</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 0.00 0.00 0.00 0.04 0.04 0.00 0.04 0.00 0.01 0.07 0.00 0.00</td>
<td>A 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.03 0.00 0.00</td>
</tr>
<tr>
<td>C 0.00 0.04 0.00 0.10 0.16 0.16 0.03 0.28 0.00 0.50 0.15 0.41</td>
<td>C 0.00 0.04 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.03 0.00 0.00</td>
</tr>
<tr>
<td>G 0.01 0.01 0.06 0.01 0.00 0.00 0.00 0.00 0.00 0.03 0.00 0.00</td>
<td>T 0.99 0.94 0.94 0.84 0.78 0.84 0.93 0.72 0.99 0.40 0.85 0.49</td>
</tr>
<tr>
<td>T 0.99 0.94 0.94 0.84 0.78 0.84 0.93 0.72 0.99 0.40 0.85 0.49</td>
<td>G 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>A 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
<td>C 0.86 0.00 0.00 0.00 0.00 0.18 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>C 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
<td>G 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>G 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
<td>T 0.14 1.00 1.00 1.00 1.00 0.82 0.93 1.00 0.71</td>
</tr>
<tr>
<td>T 0.14 1.00 1.00 1.00 1.00 0.82 0.93 1.00 0.71</td>
<td></td>
</tr>
</tbody>
</table>

### Table D.3: The motifs predicted from MEME analysis of the promoter regions of cluster A, but found in combination in the upstream promoter regions of genes in the “RNase H” protein family (figure 7.13).

<table>
<thead>
<tr>
<th>Cluster A - Motif Prediction 6</th>
<th>Consensus: GAGTGTGTGTGTGTG</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 0.07 0.53 0.27 0.13 0.20 0.00 0.27 0.00 0.00 0.27 0.07 0.00</td>
<td>A 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>C 0.00 0.00 0.33 0.20 0.00 0.07 0.00 0.00 0.13 0.20 0.00 0.00</td>
<td>C 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>G 0.93 0.00 0.40 0.07 0.80 0.27 0.53 0.00 0.87 0.07 0.73 0.07</td>
<td>G 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>T 0.00 0.47 0.00 0.60 0.00 0.67 0.20 1.00 0.00 0.73 0.00 0.87</td>
<td>T 0.00 0.47 0.00 0.60 0.00 0.67 0.20 1.00 0.00 0.73 0.00 0.87</td>
</tr>
<tr>
<td>T 0.00 0.47 0.00 0.60 0.00 0.67 0.20 1.00 0.00 0.73 0.00 0.87</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cluster A - Motif Prediction 9</th>
<th>Consensus: CCCCCCCTTCATCCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>A 0.00 0.06 0.00 0.06 0.00 0.17 0.00 0.00 0.06 0.06 0.00 0.22</td>
<td>A 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>C 0.83 0.56 0.00 0.0 1.00 0.77 0.00 0.39 0.94 0.94 0.33 0.00</td>
<td>C 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>G 0.00 0.00 0.06 0.94 0.00 0.06 0.00 0.61 0.00 0.00 0.00 0.00</td>
<td>T 0.17 0.38 0.94 0.0 0.00 0.00 1.00 0.00 0.00 0.00 0.67 0.39</td>
</tr>
<tr>
<td>T 0.17 0.38 0.94 0.0 0.00 0.00 1.00 0.00 0.00 0.00 0.67 0.39</td>
<td></td>
</tr>
</tbody>
</table>


Bibliography 200


