Bayesian Variable Selection in Clustering and Hierarchical Mixture Modeling

by

Lin Lin

Department of Statistical Science
Duke University

Date: __________________________
Approved:

Mike West, Supervisor

Jerry Reiter

Li Ma

Cliburn Chan

Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistical Science in the Graduate School of Duke University 2012.
**Abstract**

Bayesian Variable Selection in Clustering and Hierarchical Mixture Modeling

by

Lin Lin

Department of Statistical Science
Duke University

Date: _______________________

Approved: _______________________

____________________________
Mike West, Supervisor

____________________________
Jerry Reiter

____________________________
Li Ma

____________________________
Cliburn Chan

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistical Science in the Graduate School of Duke University 2012
Copyright © 2012 by Lin Lin
All rights reserved
Abstract

Clustering methods are designed to separate heterogeneous data into groups of similar objects such that objects within a group are similar, and objects in different groups are dissimilar. From the machine learning perspective, clustering can also be viewed as one of the most important topics within the unsupervised learning problem, which involves finding structures in a collection of unlabeled data. Various clustering methods have been developed under different problem contexts. Specifically, high dimensional data has stimulated a high level of interest in combining clustering algorithms and variable selection procedures; large data sets with expanding dimension have provoked an increasing need for relevant, customized clustering algorithms that offer the ability to detect low probability clusters.

This dissertation focuses on the model-based Bayesian approach to clustering. I first develop a new Bayesian Expectation-Maximization algorithm in fitting Dirichlet process mixture models and an algorithm to identify clusters under mixture models by aggregating mixture components. These two algorithms are used extensively throughout the dissertation. I then develop the concept and theory of a new variable selection method that is based on an evaluation of subsets of variables for the discriminatory evidence they provide in multivariate mixture modeling. This new approach to discriminative information analysis uses a natural measure of concordance between mixture component densities. The approach is both effective and computationally attractive for routine use in assessing and prioritizing subsets of variables according
to their roles in the discrimination of one or more clusters. I demonstrate that the approach is useful for providing an objective basis for including or excluding specific variables in flow cytometry data analysis. These studies demonstrate how ranked sets of such variables can be used to optimize clustering strategies and selectively visualize identified clusters of the data of interest.

Next, I create a new approach to Bayesian mixture modeling with large data sets for a specific, important class of problems in biological subtype identification. The context, that of combinatorial encoding in flow cytometry, naturally introduces the hierarchical structure that these new models are designed to incorporate. I describe these novel classes of Bayesian mixture models with hierarchical structures that reflect the underlying problem context. The Bayesian analysis involves structured priors and computations using customized Markov chain Monte Carlo methods for model fitting that exploit a distributed GPU (graphics processing unit) implementation. The hierarchical mixture model is applied in the novel use of automated flow cytometry technology to measure levels of protein markers on thousands to millions of cells.

Finally, I develop a new approach to cluster high dimensional data based on Kingman’s coalescent tree modeling ideas. Under traditional clustering models, the number of parameters required to construct the model increases exponentially with the number of dimensions. This phenomenon can lead to model overfitting and an enormous computational search challenge. The approach addresses these issues by proposing to learn the data structure in each individual dimension and combining these dimensions in a flexible tree-based model class. The new tree-based mixture model is studied extensively under various simulation studies, under which the model’s superiority is reflected compared with traditional mixture models.
To My Family
Contents

Abstract iv
List of Tables xi
List of Figures xiii
Acknowledgements xvii

1 Introduction 1
  1.1 Flow cytometry ............................................. 4
  1.2 Dirichlet process mixture models .......................... 6
  1.3 Clustering using Dirichlet process mixture models ........ 9
  1.4 Hierarchical Dirichlet process mixture models ............. 11
  1.5 Bayesian agglomerative clustering ......................... 12
    1.5.1 Kingman’s coalescent ................................ 13
    1.5.2 Hierarchical clustering with the coalescent .......... 14
  1.6 Variable selection methods for clustering .................. 15

2 Bayesian EM and subgroup identification in Gaussian DP mixtures 19
  2.1 Introduction ................................................. 19
  2.2 Introduction to Bayesian EM algorithm ..................... 20
  2.3 Bayesian EM algorithm in Dirichlet process mixtures ...... 22
  2.4 Non-Gaussian subgroups by aggregating Gaussian components 24
  2.5 Discussion .................................................. 26
3 Discriminative information analysis in mixture modeling 28
  3.1 Introduction ......................................................... 28
  3.2 Density concordance and discriminative information measure ... 30
    3.2.1 Definitions ..................................................... 30
    3.2.2 Theoretical insights .......................................... 32
    3.2.3 Misclassification rate connection ............................ 35
    3.2.4 Variable selection procedures ............................... 37
  3.3 Simulated data examples ......................................... 38
  3.4 Discriminatory marker variables in flow cytometry data analysis . 47
  3.5 Discussion ......................................................... 51

4 Discriminative information measure in flow cytometry data analysis 53
  4.1 Introduction ......................................................... 54
  4.2 Evaluation and “gold standards” .................................. 56
  4.3 Finding proliferating lymphocytes with Bayesian mixture modeling . 56
  4.4 Finding the optimal combination of $k$ parameters to identify target
cell subsets ............................................................. 57
  4.5 Evaluating the drop in sensitivity and specificity when using fewer
parameters ................................................................. 59
  4.6 Simplification of gating strategy ................................... 61
  4.7 Discussion .......................................................... 62

5 Hierarchical Bayesian mixture modeling 65
  5.1 Introduction ......................................................... 65
  5.2 Flow cytometry in immune response studies ....................... 67
  5.3 Hierarchical mixture modeling .................................... 69
    5.3.1 Data structure and mixture modeling issues ................. 69
    5.3.2 Hierarchical model ............................................. 72
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3.3 Mixture model for biomarkers</td>
<td>72</td>
</tr>
<tr>
<td>5.3.4 Conditional mixture models for multimers</td>
<td>74</td>
</tr>
<tr>
<td>5.3.5 Priors on multimer component location vectors</td>
<td>77</td>
</tr>
<tr>
<td>5.4 Posterior computations</td>
<td>79</td>
</tr>
<tr>
<td>5.4.1 Augmented model and MCMC</td>
<td>79</td>
</tr>
<tr>
<td>5.4.2 Update component indicator variables</td>
<td>80</td>
</tr>
<tr>
<td>5.4.3 Update biomarker model parameters</td>
<td>81</td>
</tr>
<tr>
<td>5.4.4 Update multimer model parameters</td>
<td>83</td>
</tr>
<tr>
<td>5.4.5 Post-MCMC analysis</td>
<td>85</td>
</tr>
<tr>
<td>5.4.6 Computational implementation</td>
<td>86</td>
</tr>
<tr>
<td>5.5 Simulation study</td>
<td>87</td>
</tr>
<tr>
<td>5.6 Study of data from human blood samples</td>
<td>91</td>
</tr>
<tr>
<td>5.7 Discussion</td>
<td>96</td>
</tr>
<tr>
<td>6 Bayesian tree-based mixture models</td>
<td>99</td>
</tr>
<tr>
<td>6.1 Introduction</td>
<td>99</td>
</tr>
<tr>
<td>6.2 Bayesian mixture of binary trees</td>
<td>101</td>
</tr>
<tr>
<td>6.2.1 Initial distribution specification</td>
<td>104</td>
</tr>
<tr>
<td>6.2.2 Mixture of binary trees model</td>
<td>105</td>
</tr>
<tr>
<td>6.3 Posterior computations</td>
<td>107</td>
</tr>
<tr>
<td>6.3.1 Update component indicator variables</td>
<td>108</td>
</tr>
<tr>
<td>6.3.2 Update mixture probabilities and hyperparameter of the DP prior</td>
<td>108</td>
</tr>
<tr>
<td>6.3.3 Update tree structures</td>
<td>108</td>
</tr>
<tr>
<td>6.3.4 Update mean locations</td>
<td>109</td>
</tr>
<tr>
<td>6.4 Simulation studies</td>
<td>111</td>
</tr>
<tr>
<td>6.5 Discussion</td>
<td>121</td>
</tr>
<tr>
<td>Chapter</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>7</td>
<td>Summary and future work</td>
</tr>
<tr>
<td></td>
<td>Bibliography</td>
</tr>
<tr>
<td></td>
<td>Biography</td>
</tr>
</tbody>
</table>
3.1 Example 1. Discriminative measures $d_c(h)$, Monte Carlo estimates of misclassification rates $r_c(h)$ and the estimates and estimation standard error from equation (3.2). Calculations are based on the 3-dimensional Gaussian mixture with parameters estimated as posterior modes. . . . 38

3.2 Example 2. Discriminative measures $d_c(h)$, Monte Carlo estimates of misclassification rates $r_c(h)$ and the estimates and estimation standard error from equation (3.2). Calculations are based on the 3-dimensional Gaussian mixture with parameters estimated as posterior modes. . . . 41

3.3 Example 3. Discriminative measures $d_c(h)$, for $c = 1, 2$. Calculations for $d_c(h)$ are based on the 8-dimensional Gaussian mixture with parameters estimated as posterior modes. The variable added at each step is listed with the corresponding values of $d_c(h)$ and RSM. The last variable to be included under the pre-specified stopping rules is underlined. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 44

3.4 Example 4. Discriminative measures $d_c(h)$, for $c = 1, ..., 4$. Calculations for $d_c(h)$ are based on the 8-dimensional Gaussian mixture with parameters estimated as posterior modes. The variable added at each step is listed with the corresponding values of $d_c(h)$ and RSM. The last variable to be included under the pre-specified stopping rules is underlined. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 46

3.5 Example 5. Discriminative measures $d_c(h)$, for $c = 1, ..., 4$. Calculations for $d_c(h)$ are based on the 8-dimensional Gaussian mixture with parameters estimated as posterior modes. The variable added at each step is listed with the corresponding values of $d_c(h)$ and RSM. The last variable to be included under the pre-specified stopping rules is underlined. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
3.6 Summary discriminative measures for selected marker variable subsets in the analysis of Tregs flow cytometry data, with variables: FSC-A, SSC-A, CD4, CD45RO, CD39, CD127, CD25, CD95, HLADR, FoxP3, CD152, CD3 vAmineViolet labeled from 1 to 13, respectively.

6.1 Example 1. Classification rate for the 2 estimated mixture components based on 4 consecutive MCMC outputs corresponding to Figure 6.8. The classification rate is obtained by dividing the true positive by the true size of that component.

6.2 Example 3. Classification rate for 4 major estimated components resulting from the hierarchical clustering procedure. Comp. \( j \) represents the normal component \( j \) and Estimated Comp. \( j \) is the estimated component \( j \). The classification rate is obtained by dividing the true positive by the true size of that component.

6.3 Example 3. Classification rate for 4 major estimated components resulting from the hierarchical clustering procedure. Comp. \( j \) represents the normal component \( j \) and Estimated Comp. \( j \) is the estimated component \( j \). The classification rate is obtained by dividing the true positive by the true size of that component.

6.4 Example 3. Classification rate for 4 major estimated components resulting from the mixture of trees model. Comp. \( j \) represents the normal component \( j \) and Estimated Comp. \( j \) is the estimated component \( j \). The classification rate is obtained by dividing the true positive by the true size of that component.
List of Figures

3.1 Plot of $S(\beta)$ (black line) and $1 - d_\beta$ (dashed line) for $\beta$ ranging from 2.5 to 10. ............................................. 34

3.2 Plot of the first derivative of $S(\beta)$ (black line) and $1 - d_\beta$ (dashed line) for $\beta$ ranging from 2.5 to 10. ....................... 35

3.3 Pairwise scatter plots of a randomly selected subset of the $n = 5,000$ observations in Example 1. Dimensions 1 and 2 together discriminate the 2 normal components while dimension 3 is redundant. ............. 39

3.4 Posteriors for $d_1(1, 2, 3)$, left, and $d_1(1, 2)$, right, in Example 1. The posteriors are truncated at higher values for clarity. Dotted lines indicate posterior means. The $d_\gamma(h)$ estimates based on mixture model parameters set at posterior modes are indicated by the dark dashed vertical lines. .................................................. 40

3.5 Pairwise scatter plots of a randomly selected subset of the $n = 5,000$ observations in Example 2. ................................. 41

3.6 Pairwise scatter plots of a randomly selected subset of the $n = 6,000$ observations in Example 3. ................................. 43

3.7 Pairwise scatter plots of a randomly selected subset of the $n = 6,000$ observations in Example 4. ................................. 45

3.8 Pairwise scatter plots of a randomly selected subset of the $n = 6,000$ observations in Example 5. ................................. 47
3.9 Scatter plots of a small random subsample of the flow cytometry data on some important marker variable dimensions. The contours are those of the 2-dimensional margins on identified subpopulations $f_c(x)$ in the full 13-dimensional mixture; the area of each contour displayed is approximately proportional to corresponding estimated probabilities $\alpha_c$. This is based on the model with parameters estimated by the posterior mode, and non-Gaussian subtypes are defined using the mixture aggregation method as described in Section 2.4.

3.10 Histogram of posterior samples for $d_T(1:13)$ in the analysis of flow cytometry data, with the modal estimate indicated by the vertical dashed line. Dotted line indicates posterior mean.

4.1 Modal clusters identified as CD4 non-proliferating (26), CD4 proliferating (29), CD8 non-proliferating (25) and CD8 proliferating (30) T lymphocytes obtained by clustering on all 8 dimensions.

4.2 A schematic showing the parameter subsets with the best discriminative ability for each target cell subset.

4.3 A schematic showing the parameter subsets with the least discriminative ability for each target cell subset.

4.4 The effectiveness of DIME was checked by exhaustively calculating the sensitivity, specificity and accuracy of every parameter combination.

4.5 Design of a simplified 3-generation gating strategy using 5 parameters suggested by use of DIME and cell subsets identified with mixture model analysis. In each step, the red labeled clusters were discarded and yellow labeled clusters retained.

4.6 Comparison of simplified gating strategy (red symbols) with the original 9-generation gating strategy (blue symbols) on all the CFSE data sets used in a single experiment across 3 different labs (BD=square, Duke=triangle, NIML=circle) under 3 different treatment conditions (unstimulated, CMVpp65 peptide mix (15mers overlapping by 11aa) and anti-CD3/CD28 mAbs).

5.1 3D scatter plots of a randomly selected subset of the FCM data of Section 5.6 on 3 reporters. Left: Qdot 655-A vs. Qdot 605-A vs. PE-A. Right: Qdot 655-A vs. APC-A vs. PE-A.

5.2 Data of Figure 5.1 on contours of 2-dimensional margins of fitted 10-dimensional DP Gaussian mixture.
5.3 Pairwise scatter plots and one 3-dimensional scatter plot of simulated data. The cluster of cells of interest is plotted in red. Here \(b^i\) is the \(i\)th dimension in the biomarker subvector, and \(t^i\) is the \(i\)th dimension in the tetramer subvector. The lower right plot indicates that the cluster of interest is the outer layer of the sample under \(b^2\), \(t^1\) and \(t^2\). The other 3 subplots shows that the cluster is hard to identify with traditional 2-D gating.

5.4 Scatter plots of synthetic data example as in Figure 5.3. Using standard DP mixture analysis, a relatively large subtype is identified. Cells assigned to this subtype are colored blue, and this blue region extends to include much of the actual subtype region, in red, in this synthetic data set. The analysis is unable to identify the correct subtype region.

5.5 Trace plot over the last 5,000 MCMC iterates of computed values of the model of equation (5.1) on seven randomly selected data points.

5.6 Scatter plots of synthetic data example as in Figures 5.3 and 5.4. The cluster of interest is plotted in red, and almost wholly overlays that identified by the hierarchical mixture model in blue.

5.7 Reference plot indicating clusters/subtypes of interest in the T-cell human blood data. A subset of the data is shown in grey, CD8+ T cells are plotted in yellow, CMV, EBV and FLU groups are plotted in green, red, and blue, respectively. These latter three identified subtypes contained 2739, 66 and 59 observations respectively. Subtypes of interest are CD8+ and positive also in the three corresponding multimer features.

5.8 Trace plot over the last 1,000 MCMC iterates of model density values on five randomly selected data points in analysis of human blood data.

5.9 Scatter plots of the real data. The contours are the 2-dimensional margins on each Gaussian component in the full 10-dimensional mixture.

5.10 Identified clusters of interest. A subset of the data is shown in grey. CMV, EBV and FLU groups are plotted in green, red, and blue respectively. The identified three groups contain 2849, 4 and 216 observations respectively. The group with 4 events does not meet the cut-off of 10 events and 0.002% of CD8 for positive events we have suggested (Andersen et al., 2012) and may represent false positive or background events.

6.1 Segment of the binary tree structure
6.2 3-dimensional and 2-dimensional plots of log-likelihood (6.2) values under flat initial distribution, with \( z_{p-1} \sim N(0, 100) \). .......................... 105

6.3 3-dimensional and 2-dimensional plots of log-likelihood (6.2) values for the same tree and \( \mu \) in Figure 6.2 under informative initial distribution, with \( z_{p-1} \sim N(0, 0.5) \). .................................................. 106

6.4 Pairwise scatter plots of the generated data set in Example 1. .......................... 112

6.5 Trace plot for \( \mu_{s,j} \) under the last 1000 iterations, \( s = 1, 2, j = 1 : 5 \) .......................... 113

6.6 The estimated tree structures from the last MCMC iteration in Example 1. Color edges are for easy visualization. ........................................... 114

6.7 Pairwise scatter plots of the generated data set in Example 1. Colors serve as component indicators obtained based on the last MCMC iteration. ........................................... 115

6.8 The estimated tree structures from four consecutive MCMC iterations in Example 1. Each row represents one MCMC iteration and columns represent two components. ........................................... 116

6.9 Pairwise scatter plots of the first 6 dimensions of the generated data set in Example 2. ........................................... 117

6.10 The estimated tree structures in Example 2 based on the last MCMC iteration ........................................... 118

6.11 Pairwise scatter plots of the first 5 dimensions of the generated data set in Example 3. ........................................... 120

6.12 The estimated tree structures in Example 3 based on the last MCMC iteration ........................................... 122
Acknowledgements

I would like to express my deep and sincere gratitude to everyone who supported and helped me over the past few years. They have made my graduate study possible.

First and foremost, I would like to thank my advisor Mike West for his insightful guidance, perpetual encouragement, endless patience and invaluable advice throughout my doctoral study. Working with Mike has been a great pleasure for me; his constant support and inspiration make my doctoral research productive and stimulating. Mike is always enthusiastic and knowledgeable about my work, I have benefited intellectually from his piercing insights and thoughtful critiques for good research. I am also thankful for the excellence role model he has provided me as both an influential research leader and a terrific mentor.

I wish to thank my thesis committee members, Jerry Reiter, Li Ma and Cliburn Chan for their time, support and valuable comments in creating this dissertation. I am especially grateful to Cliburn for his continuous support on the Flow Cytometry projects which inspired me with ideas for the dissertation. I am also thankful to Fan Li, who was my first year advisor. Her kindness and help allowed me to make a smooth transition to life at Duke.

The faculty of this department are very resourceful and have provided me with a tremendous graduate education. Further thanks go to Alan Gelfand, Robert Wolpert, David Dunson, Fan Li, Scott Schmidler, Dalene Stangl, James Berger, and Merlise Clyde for their enlightenment, which has provided me a solid foundation to progress
upon in my graduate study. I also would like to thank Quanli Wang for his help in GPU programming that made my Matlab code enormously fast.

The department of statistical science at Duke university is a great home. Friends such as Simon Lunagomez, Chiranjit Mukherjee, Chunlin Ji, Hao Wang, Danilo Lopes, Scott Schwartz, Anirban Bhattacharya, Debdeep Pati, Francesca Petralia, Andrew Cron, Kai Cui, Jianyu Wang, Xiaojing Wang, Yajuan Si, and many others have enriched my graduate study experience. I am also privileged to work with Ricardo Henao, who has patiently acquainted me with his work on coalescent clustering.

Last, but most importantly, I am immensely indebted to my parents, Jianzhang Lin and Wenyi Xiao. Words can not express how grateful I am for their nurture, upbringing and unconditional love. Likewise, I am deeply grateful to my boyfriend, Xingyuan Fang, for his incredible support, encouragement and love. To them I dedicate this dissertation.
The research presented in this dissertation focuses on using Bayesian model-based techniques for clustering, particularly in multivariate mixture models. Its central focus is Bayesian variable selection in clustering and hierarchical mixture modeling. I am particularly concerned with the situation of increasingly high dimensional data sets with larger numbers of variables and/or observations where the numbers of mixture components representing meaningful subpopulations can be large. Traditional mixture models have limited mechanism to differentiate subsets of variables that play roles in discriminating of subpopulations; these models also encounter difficulties identifying low probability component structures when fitting large data sets requiring many mixture components. Hence, it is essential to develop new statistical approaches to analyze this type of data set. A good example of the need for a variable subset assessment method can be observed in routinely applied biological cell assays using flow cytometry, which generate multiple, large data sets in 10 – 20 dimensions (e.g. Chan et al., 2008; Finak et al., 2009). The use of automated mixture model fitting and a subsequent discriminatory analysis to prioritize subsets of variables will have a major impact on advancing statistical work in these and other areas. A good
example of the need for hierarchical mixture modeling can be observed in the novel use of automated flow cytometry technology to measure levels of protein markers on thousands to millions of cells (Hadrup et al., 2009). Structured, hierarchical mixture models with the applied goals of automated inference to identify specific cellular subtypes in very large samples of T-cells offer advantages in identifying and quantifying subpopulation structures related to relatively rare cell subtypes.

Multiple, varying methods and approaches have been developed for variable selection in clustering, including wrapper and distance-based approaches (Dy and Brodley, 2004; Friedman and Meulman, 2004) and mixture model-based approaches (Kim et al., 2006; Raftery and Dean, 2006; Law et al., 2004). In contrast to these approaches, I address the discriminatory analysis following model fitting. First, I assume that the model is available with either estimated plug-in parameters or Markov chain Monte Carlo based posterior samples (e.g. Lavine and West, 1992; West, 1992, 1997; Richardson and Green, 1997); I then interrogate the model to define the variable subset discriminatory measures. Furthermore, I directly address the issue that very different subsets of variables may play roles in differentiating different mixture components. In addition, I define computationally effective approaches to enable access and routine use as well as the ability to scale to larger models.

I also extend traditional Bayesian mixture models to a novel class of structured, hierarchical mixture models under large data sets with high dimensions. This approach involves a natural, model-based hierarchical partitioning of the variables and results in efficient and precise mixture modeling analysis applied to smaller subsets of data across the hierarchy, while the inclusion of structured priors induce a focus on the relevant parameter regions of interest. This process generates fitted models in which low probability mixture components are appropriately located in weakly populated regions of the sample space. Compared with the standard mixture approaches, the new model, by design, is able to identify and quantify relatively rare
Finally, I present a new, general approach to clustering in high dimensional data. Unlike previous mixture model-based approaches, which involve estimating the covariance matrix for each Gaussian component, or graphical model-based clustering, which aims to infer variable structures based on statistical dependencies and/or encourage a sparse covariance matrix, the focus here is on modeling data starting from low dimensions to avoid the curse of dimensionality. A tree structure is placed on the variables such that each leaf node represents one dimension, to infer variable structures based on their relative distances. This tree structure can be used as an approximation of the covariance matrix requiring significantly fewer parameters. The data density is, therefore, deduced from the tree structure.

This dissertation is organized as follows. In this chapter, I first introduce motivations to my dissertation, then provide a broad overview of the nonparametric Bayesian methods relevant to the research developed here and of Bayesian variable selection methods used in clustering analysis. Beginning in Section 1.1, I provide an introduction to flow cytometry. Section 1.2, I describe the Dirichlet process mixture (DPM) models. Section 1.3 explains the clustering mechanisms under DPM models. Section 1.4 describes a hierarchical version of the DPM model, followed by an overview of Bayesian agglomerative clustering with coalescents – a special class of Bayesian hierarchical clustering models – in Section 1.5. In section 1.6, I review some of the existing methods for variable selection in clustering.

In Chapter 2, I present a new Bayesian Expectation-Maximization algorithm for fitting DPM models and an algorithm for finding clusters in DPM models. In Chapter 3, I introduce my method for variable selection in clustering, explore the effectiveness of the method on simulated data, and examine its application to a large flow cytometry data set in high-throughput systems biology. In Chapter 4, I present a detailed application of variable selection to flow cytometry data analysis.
Chapter 5, I develop a structured, hierarchical mixture model for identifying and quantifying subpopulation structures related to relatively rare subtypes. The performance of this model is assessed using simulated data and a data set from an experimental study of antigen-specific T-cell subtyping using combinatorially encoded assays in human blood samples. In Chapter 6, a new approach to clustering high dimensional data is developed, with an emphasis on avoiding the curse of dimensionality and increasing model fitting efficiency. This approach defines a natural, hierarchical tree-based modeling framework. Finally, in Chapter 7, I conclude with a summary of the work and a discussion of the potential next steps and open problems.

1.1 Flow cytometry

The ability to monitor complex immune responses quantitatively is essential for the development of effective vaccines and the discovery of diagnostic or prognostic biomarkers for clinical trials. Multi-parameter flow cytometry is an ideal, sample-sparing assay for constructing informative profiles of immune response, as it can measure multiple immune parameters (cell phenotype, activation or maturation status, intracellular cytokine or other effector molecule concentrations) with a single sample, and provides a detailed snapshot of the immune response that is ideal for profiling. Historically, the blood cells were evaluated manually using microscope. The flow cytometer, which was invented by Mack Fulwyler (Fulwyler, 1965), made a further advancement by combining optical and computer techniques to automatically produce a tremendous amount of data within a very short period of time. An introduction for flow cytometry can be found, e.g. Shapiro (2005). In general, a flow cytometer consists of three components: fluidics, optics and electronics. The cell sample stained with one or more fluorescent dyes specific to cellular label components of interest is first suspended in a stream of fluid. Cells are then passed, one by one, through a laser beam. A set of detectors are placed in order to detect light sig-
nals when the stream of cells passes through the light beam: forward-scattered light (FSC), which is proportional to the surface area of size of a cell, and side-scattered light (SSC), which is proportional to the granularity or internal complexity of a cell, are captured by detectors placed in line with the light beam and perpendicular to it, respectively. The rest are fluorescence detectors. The fluorochrome found in the cell or attached to the cell may be excited into emitting light at a longer wavelength than the light source; this can then be optically separated through optical filters. The electronics system converts the detected light signals into electronic signals that can be processed by the computer. Hence, the characteristics of each cell are based on its light scattering and fluorescent intensities.

One of the main objectives in using flow cytometry data is to identify different cell subpopulations, usually of very low probabilities around 0.01%, from a large set of data. Standard analysis has relied on a manual strategy that filters cells via serial 2D projections of reporter space using visually defined boundaries known as gates. The process of gating relies heavily on local expertise, and is laborious, error-prone, and cumbersome in higher dimensions since the number of possible 2D projections that need to be examined increases rapidly. This poses a bottleneck in the use of higher dimensional data for cell identification. This partly underlies the drive to automatic cell subset identification to overcomes the limitations of manual gating, and the increasing adoption of statistical mixture modeling approaches (e.g. Chan et al., 2008; Finak et al., 2009; Lo et al., 2008; Pyne et al., 2009). Bayesian mixture models have demonstrated their ability in reproducing and accurately quantifying cell subset populations. However, there is an increasing need to design and implement novel nonparametric Bayesian models that can flexibly and robustly fit data with an unknown number of mixture components efficiently, in addition to handling non-Gaussian cell subtype distributions robustly. Models are also of interest to automatically prioritize markers that are useful for identifying a given cell subtype,
simplify gating strategy design, reduce assay development time, and make it easier for multiple institutions to use the same protocols. Computational challenges often arise in fitting mixture models with data sets of increasing heterogeneity and larger number of observations, which require increasingly high-dimensional mixture models with large numbers of mixture components. Thus, strategies such as parallelization and GPU implementation in Bayesian posterior simulation offers many potential benefits. The other challenge arises in identifying low probability component structure in fitting large data sets requiring many mixture components; the inherent masking issue makes the mixture model difficult to discover and quantify inferences on the biologically interesting but small clusters that deviate from the bulk of the data. To adequately estimate low density regions would require a huge increase in the number of mixture components and an enormous computational search challenge, and is simply infeasible as a routine analysis. This gives rise to the need for more structured, hierarchical mixture modeling strategies to efficiently and accurately quantify extremely low frequency clusters in very large data sets.

Hence, I focus on developing theory and implementing efficient algorithms for new nonparametric mixture models that handle non-Gaussian clusters; discriminative information analyses methods to automatically prioritize variables that are useful for identifying a given cluster; and structured, hierarchical mixture models to efficiently and accurately quantify extremely rare events in very large data sets and cluster high-dimensional data sets.

1.2 Dirichlet process mixture models

Bayesian nonparametric methods have been studied and developed extensively; these methods have also been advanced by the development of Markov chain methods for posterior sampling. Dirichlet processes (DPs), which can be understood as distributions over distributions, were first formalized by Ferguson (1973) and Antoniak
(1974). Since then, DPs have played an important role in nonparametric Bayesian methods. Many papers have been devoted to developing the practicality of using DP priors, such as those by Sethuraman (1994), Escobar and West (1995), MacEachern and Müller (1998) and Neal (2000).

Consider a set \( \Phi \) and a partition \( \{A_1, A_2, \ldots \} \) of \( \Phi \) such that \( \bigcup A_k = \Phi \). Let \( G \) be a random probability measure on \( \Phi \) such that for all \( i \), \( G(A_i) \) is a random variable. Ferguson (1973) showed that \( G \) is a DP with parameters \( \alpha \) and \( G_0 \), denoted by \( G \sim DP(\alpha, G_0) \), if for any partition \( \{A_1, \ldots, A_k\} \), the vector of random probabilities \( G(A_k) \) follows a Dirichlet distribution, i.e.

\[
(G(A_1), \ldots, G(A_k)) \sim \text{Dirichlet}(\alpha G_0(A_1), \ldots, \alpha G_0(A_k)).
\]

The base distribution \( G_0 \) has a parametric form and \( E(G(A)) = G_0(A) \). \( G_0 \) acts as a prior distribution for component parameters in the DPM models discussed later in this section. The concentration parameter \( \alpha \), a positive scalar, controls the variance of the DP. As \( \alpha \) increases, a sample \( G \) is more likely to be close to \( G_0 \), i.e., \( G \to G_0 \) as \( \alpha \to \infty \). Thus, \( \alpha \) represents degrees of confidence in the base measure \( G_0 \).

Sethuraman (1994) provides a constructive approach, the stick-breaking construction, to generate a DP. Let \( \{u_k\}_{k=1}^\infty \) be a set of independent beta distributed random variables such that \( u_k \sim Be(1, \alpha) \) and \( \theta_k \sim G_0 \), for \( k = 1, 2, \ldots \). Often \( G_0 \) is continuous so that the \( \theta_k \) are distinct with probability one. If \( G \sim DP(\alpha, G_0) \), then the stick-breaking representation of \( G \) is as follows:

\[
\pi_k = u_k \prod_{j=1}^{k-1} (1 - u_j),
\]

\[
G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k},
\]

where \( \delta_{\theta_k} \) is a point mass at \( \theta_k \). This construction emphasizes that the samples from a DP are discrete with probability one. The term “stick-breaking” comes from the
interpretation of mixing proportion $\pi_k$, which is given by successively breaking a unit-length stick into infinitely many pieces. After portions are assigned to the first $k - 1$ values, the size of the next value is given by an independent draw $u_k$ from $Be(1, \alpha)$, proportional to the length of the remainder of the stick, $\prod_{i=1}^{k-1}(1 - u_i)$. On average, a smaller $\alpha$ corresponds to a shorter remaining stick for the subsequent values.

To link DP to nonparametric Bayesian modeling, now consider a hierarchical model in which observations $x_i$ are sampled from a parametric distribution $f$ with parameter $\theta_i$. The model can be described as follows:

$$x_i | \theta_i \sim f(x_i | \theta_i),$$

$$\theta_i \sim G.$$

A parametric distribution can be assigned as the prior distribution $G$. However, to ensure greater model flexibility and robustness, a DP prior can be placed on the parameter distribution, $G \sim DP(\alpha, G_0)$. The general Bayesian hierarchical model with a DP prior can then be written as:

$$x_i | \theta_i \sim f(x_i | \theta_i),$$

$$\theta_i \sim G,$$

$$G \sim DP(\alpha, G_0).$$ (1.1)

This is called the DPM model. According to the discreteness property of the DP and its stick-breaking representation, this model implies $x_i \sim \sum_{k=1}^{\infty} \pi_k f(x_i | \theta_k)$, where the $\theta_k$ are an infinite samples from $G_0$. Under this formulation, the DPM is interpreted as a flexible mixture model in which the number of components is infinite.

In summary, a hierarchical Bayesian model with a DP prior leads to a DPM model (1.1). This model allows different observations to be associated with the same underlying component, making it suitable for model-based clustering.
1.3 Clustering using Dirichlet process mixture models

Partitioning and mixture models are widely used to define and estimate cluster structures. There are two classes of clustering methods: hierarchical schemes producing a hierarchical sequence of partitions, a particular class of which will be explained in Section 1.5, and allocation schemes in which observations are allocated among the proposed clusters. The latter are based on the concept that the observations come from a heterogeneous population consisting of several clusters. Each cluster can be modeled by a distinct parametric distribution and a mixture of these clusters, a finite mixture model, is used to model the heterogeneity of the overall population. More explicitly, given observations $x = \{x_1, \cdots, x_n\}$, assume first that there are $K$ clusters differentiated by $\theta_k$, which can be either a scalar or a vector. Let $f(x|\theta_k)$ be the density of cluster $k$; the finite mixture model is, then, of the form

$$f(x|\theta) = \sum_{k=1}^{K} \pi_k f(x|\theta_k)$$

with the mixing proportions $\pi_k \geq 0$, for $k = 1, \cdots, K$, and $\sum_{k=1}^{K} \pi_k = 1$. The probability $\pi_k$ also represents the prior probability that an observation comes from each cluster $k$. One of the difficulties of using finite mixture model in clustering analysis is in determining the number of clusters $K$.

Compared with the finite mixture model, the use of the DPM model becomes particularly attractive because it does not require specifying $K$ a priori. Using the stick-breaking representation, $G$ can be written as $\sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$. Let $z_i$ denote the latent configuration indicator with $P(z_i = k) = \pi_k$. The DPM model (1.1) can be
equivalently expressed as

\[ x_i \sim f(x_i | \theta_{z_i}), \]

\[ P(z_i = k) = \pi_k, \]

\[ \pi_k = u_k \prod_{l=1}^{k-1} (1 - u_l), \]

\[ u_k \sim Be(1, \alpha), \]

\[ \theta_k \sim G_0, \]

for \( i = 1, \cdots, n \) and \( k = 1, 2, \ldots \). In posterior inference, both the mixture component identities (i.e. \( \theta_k \)) and the assignment of \( x_i \) to the components (i.e. \( z_i \)) are inferred. Hence, the idea of clustering is inherent. Although the DPM model is a mixture model with an infinite number of components, only a finite number of components are effective in a finite data set, and is approximately to the order of \( \alpha \log n \). To cluster continuous data, Dirichlet process Gaussian mixtures are widely used, defined by taking \( f_k(x | \theta_k) \) as the normal \( N(\mu_k, \Sigma_k) \), with \( \theta_k = (\mu_k, \Sigma_k) \).

Ishwaran and James (2001) developed a truncated version of the DP; where \( G_J = \sum_{k=1}^{J} \pi_k \delta_{\theta_k} \), and \( G_J \) converges almost surely to \( DP(\alpha, G_0) \) as \( J \to \infty \). \( G_J \) is defined by selecting the mixing weights \( \pi_k \) using the stick-breaking construction:

\[ \pi_k = u_k \prod_{j=1}^{k-1} (1 - u_k), \]

for \( k = 1, \ldots, J \), and \( u_k \), for \( k = 1, \cdots, J - 1 \) are independent \( Be(1, \alpha) \) random variables; then, \( u_J = 1 \) to ensure that \( \sum_{k=1}^{J} \pi_k = 1 \). The truncation \( G_J \) is computationally attractive for more efficient statistical simulation algorithms. Ishwaran and James (2001) and Ishwaran and James (2002) developed a blocked Gibbs sampling for MCMC based posterior inference under the truncated DPM model. In practice, a large value of \( J \) can be set under which certain mixture probabilities may be zero, allowing the model to reduce to fewer components than the upper bound \( J \) to the extent that is relevant to the data set at hand. The truncated version of DP is most frequently used in this dissertation, as it enables effi-
cient parallel computations for obtaining posterior inferences. Many different Markov chain Monte Carlo sampling techniques have been developed for making posterior inferences from DPM models; Neal (2000) is a good reference.

1.4 Hierarchical Dirichlet process mixture models

Rather than clustering a set of data into groups, the hierarchical Dirichlet process (HDP) mixture model (Teh et al., 2006) applies to problems of clustering grouped data. If the data are organized into a set of groups and assumed to be exchangeable both within each group and across groups, then the HDP mixture model captures the cluster structure within each group, and allows cluster identities to be shared among the groups. Explicitly, let $j$ denote the group index and $i$ denote the observation index within each group. Let $x_{j} = (x_{j1}, x_{j2}, \cdots)$ represent all the observations in group $j$. The hierarchical model is defined by the 1-way layout

$$x_{ji} \sim F(x_{ji} | \theta_{ji}),$$

$$\theta_{ji} \sim G_{j}.$$ 

The model is completed by the hierarchical specification $G_{j} | \alpha_{0}, G_{0} \sim DP(\alpha_{j}, G_{0})$, where $G_{0}$ is a global random probability measure distributed as $DP(\gamma, H)$. The $\alpha_{j}$ controls dispersion of the $G_{j}$ around the global, underlying distribution $G_{0}$. The baseline distribution $H$ provides the marginal prior distribution for $\theta_{ji}$. The distribution $G_{0}$ varies around the prior $H$, with the amount of variability controlled by $\gamma$. The atoms (parameters) in the stick-breaking representation of $G_{0}$ will be shared among all the $G_{j}$, while the parameters are also shared within each group.

The sharing of atoms among all groups in the HDP model can be made more
explicit via the stick-breaking construction:

\[ \theta_k \sim H, \]
\[ \beta'_k \sim \text{Be}(1, \gamma), \]
\[ \beta_k = \beta'_k \prod_{l=1}^{k-1} (1 - \beta'_l), \]
\[ G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k}, \]
\[ \pi'_{jk} \sim \text{Be}(\alpha_0 \beta_k, \alpha_0(1 - \sum_{l=1}^{k} \beta_l)), \quad (1.3) \]
\[ \pi_{jk} = \pi'_{jk} \prod_{l=1}^{k-1} (1 - \pi'_{jl}), \]
\[ G_j = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k}, \]
\[ z_{ji} \sim \pi_j, \]
\[ x_{ji}\mid z_{ji}, (\theta_k)_{k=1}^{\infty} \sim f(x_{ji}\mid \theta_{z_{ji}}), \]

where \( z_{ji} \) is the latent indicator with \( P(z_{ji} = k) = \pi_{jk} \). The model (1.3) implies that \( \theta_k \) are shared throughout the hierarchy.

1.5 Bayesian agglomerative clustering

In contrast to clustering using DPM models, hierarchical clustering methods seek to build a hierarchy of clusters. The strategies fall into two types: the agglomerative (bottom-up) approach, in which each observation starts in its own singleton cluster, and pairs of clusters are merged moving up the hierarchy; and the divisive (top-down) approach, in which observations start in a common cluster, and clusters are successively split moving down the hierarchy. Many methods have been developed using probabilistic models for hierarchical clustering (Neal, 2003; Williams, 1999;
Heller and Ghahramani, 2005; Teh et al., 2008). For example, Teh et al. (2008) proposed a Bayesian hierarchical clustering model that used Kingman’s coalescent as a prior over trees followed by a distribution over data points conditioned on a tree.

1.5.1 **Kingman’s coalescent**

Kingman’s coalescent is a standard model in population genetics to reconstruct the common genealogy of a countably infinite set of individuals backwards in time (Kingman, 1982a,b). Kingman’s coalescent which is a limiting distribution of $n$-coalescent, is a particular mathematical process joining lineages into common ancestors. Mathematically, the $n$-coalescent defines a prior over a binary tree consisting of $n$ leaves, one for each individual, and $n-1$ coalescent points. Thus, two elements are essential to the coalescent trees: the pairs of descendants to be merged at the $k$th coalescent point. Pairs of this kind are often enclosed in parentheses, for example, let $(\rho_{lk}, \rho_{rk})$ be the pair to merge at the $k$th coalescent point. And the branch lengths, or the merging times that each $n-1$ coalescents occurs. Specifically, let $\pi$ represent the space of partition, and $t$ denote the $n-1$ merging times. The model evolves backwards in time starting from the “present” time, $t_0 = 0$, and $\pi_0 = \{\{1\}, \cdots, \{n\}\}$. At time $t_k$, where $t_k < t_{k-1}$, $\pi_k = \{\pi_{k-1}\setminus(\rho_{lk}, \rho_{rk}), \{\rho_k\}\}$ with $\rho_k = \rho_{lk} \cup \rho_{rk}$. This merging process continues until all individuals are joined into one set. Under the $n$-coalescent, each pair of lineages are assumed to merge independently with rate 1. Thus, under the $k$th coalescence, $\Delta_k = t_{k-1} - t_k$ follows an exponential distribution with rate $(n-k+1)(n-k)/2$, as $((n-k+1)(n-k)/2)$ merges are possible at stage $k$. Then set $t_k = t_{k-1} - \Delta_k$. Two sets of $\pi_{k-1}$ will be merged uniformly into $\pi_k$. Hence, by combining choices of merges and the probabilities of durations, the probability of $\{\pi, t\}$ is as follows:

$$p(\pi, t) = \prod_{k=1}^{n-1} \exp\left(-\frac{(n-k+1)(n-k)\Delta_k}{2}\right) \tag{1.4}$$

13
The \( n \)-coalescent has certain interesting statistical properties (Kingman, 1982a). For example, the marginal distribution of \( \pi \) is uniform and independent of \( t \). And the marginal distribution of \( \pi \) is exchangeable in the set of partitions \( \pi_k \) for every \( k \).

### 1.5.2 Hierarchical clustering with the coalescent

According to equation (1.4), considering the \( n \)-coalescent as a prior over partition trees for clustering is natural. Bayesian hierarchical clustering results from modeling observed data using a Markov process evolving forward in time along the tree. Let \( x_1, \ldots, x_n \) be \( n \) observed data points at the leaves of a tree drawn from the \( n \)-coalescent. Let \( t_{lk} \) and \( t_{rk} \) be the times at which \( \rho_{lk} \) and \( \rho_{rk} \) are themselves formed.

A continuous-time Markov process evolving along the tree from the past to the present can be constructed. Let \( q(z) \) be the initial distribution of the Markov process at time \( t = -\infty \), and \( k_{st}(x, y) \) be the transition probability from state \( x \) at time \( s \) to state \( y \) at time \( t \). Often, the transition kernel is chosen as Gaussian for continuous data to allow mathematical tractability. In addition, let \( y_{\rho_i} \) be a latent variable that assumes the value of the Markov process at \( \rho_i \) just before it branches. Hence, \( y_{\rho_i} = x_i \) at leaf \( i \). The joint probability over the latent variables and the observations is as follows:

\[
p(x, y, z | \pi, t) = q(z) k_{-\infty \pi_{n-1}}(z, y_{\pi_{n-1}}) \prod_{i=1}^{n-1} k_{t_{li} t_{ri}}(y_{\rho_i}, y_{\rho_{\pi_i}}) k_{t_{li} t_{ri}}(y_{\rho_i}, y_{\rho_{\pi_i}})
\]

Teh et al. (2008) demonstrated that when using an upward pass of belief propagation on the tree, the latent variables \( \{y_{\rho_i}\}_{i=1}^{n-1} \) and \( z \) can be efficiently integrated out. Hence, the marginal probability \( p(x | \pi, t) \) can be written by a product of normalization constants:

\[
p(x | \pi, t) = Z_\pi(x, \theta_{n-1}) \prod_{i=1}^{n-1} Z_{\rho_i}(x, \theta_i),
\]
where $Z_{\rho_i}^{-1}(x, \theta_i)$ is the normalizing constant of the messages

$$M_{\rho_i}(y) = Z_{\rho_i}^{-1}(x, \theta_i) \prod_{b=l,r} \int k_{t,t_b}(y, y_b) M_{\rho_i}(y_b) dy_b.$$ 

By multiplying the prior (1.4), the joint density for the tree and observations $x$ is:

$$p(x, \pi, t) = Z_{-\rho}(x, \theta_{n-1}) \prod_{i=1}^{n-1} \exp(-(n - i + 1)(n - i)/2\Delta_i) Z_{\rho_i}(x, \theta_i).$$

A sequential Monte Carlo algorithm can then be designed to approximate the posterior using a weighted sum of point masses. Henao and Lucas (2012) later developed a new sequential Monte Carlo sampler for the coalescent-based Bayesian hierarchical clustering; this method will be elaborated and used in Chapter 6.

1.6 Variable selection methods for clustering

In high dimensional data sets, the underlying structure of interest may often be relevant to a small subset of variables, and the inclusion of irrelevant variables may degrade the model performance. Many methods have been developed for variable selection in regression and classification or supervised learning problems (e.g., George and McCulloch, 1997; Liang et al., 2008). However, few contributions have been made in the context of clustering because no prior/observed cluster labels are available to guide the variable selection. In general, there are two basic approaches to explicit variable selection for clustering: selecting variables that minimize a certain criterion, or selecting variables under a certain statistical model.

Dy and Brodley (2004) explored the wrapper framework (Kohavi and John, 1997) for unsupervised learning, which performs variable selection and clustering simultaneously. The wrapper approach divides the task into three components: variable search, clustering algorithm and variable subset evaluation. The basic idea is to search through the variable subset space, evaluating each candidate subset, by first
clustering in that subset using the clustering algorithm and then evaluating the resulting clusters and variable subset using a selected variable selection criterion. This process is repeated until a “best” variable subset with its corresponding clustering is found based on the variable evaluation criterion.

A finite Gaussian mixture model in combination with the Expectation-Maximization algorithm is used for clustering. Two variable selection criteria are investigated: the scatter separability criterion and the maximum likelihood criterion. One typically desired property is cluster separation. For \( p \)-dimensional data \( x \), let \( S_w \) denote the within-class scatter matrix and let \( S_b \) denote the between class scatter matrix, defined as

\[
S_w = \sum_{j=1}^{k} \pi_j E((x - \mu_j)(x - \mu_j)^t | \omega_j) = \sum_{j=1}^{k} \pi_j \Sigma_j,
\]

\[
S_b = \sum_{j=1}^{k} \pi_j (\mu_j - M_0)(\mu_j - M_0)^t,
\]

\[
M_0 = E(x) = \sum_{j=1}^{k} \pi_j \mu_j,
\]

where \( \pi_j \) is the probability that an observation belongs to cluster \( \omega_j \), \( k \) is the number of clusters, \( \mu_j \) is the sample mean vector of cluster \( \omega_j \), \( M_0 \) is the total sample mean and \( \Sigma_j \) is the sample covariance matrix of cluster \( \omega_j \). \( S_w \) measures how scattered the samples are from their cluster means; \( S_w \) should be as small as possible. \( S_b \) measures how scattered the cluster means are from the total mean; \( S_w \) is expected to be larger for cases of “good” discrimination. Hence trace\((S_w^{-1}S_b)\) is large in cases of good cluster discrimination. Under the maximum likelihood criterion, the mixture model is estimated via the expectation-maximization to define the maximum likelihood parameter estimates, and this trace measure serves as a criterion for both clustering and variable selection. As the exhaustive search of all possible \( 2^p \) subsets of the
variables is computationally intractable under larger $p$, a sequential forward search strategy is used.

Friedman and Meulman (2004) also developed a method of simultaneously performing variable selection and clustering using a different approach. Under their method, the selected variable subsets for each cluster may differ. The objective function of their paper is to minimize the total within-cluster separability over both the data clustering and the weights on the variables. The weights are allowed to depend on the cluster labels. Variables with high weight values are relevant to the corresponding clusters. That is, under a set of $N$ $p$-dimensional observations $x_i = (x_{i1}, \ldots, x_{ip})$, an “encoder” function $c(i)$ is defined that maps each observation $i, i = 1, \ldots, N$, to a particular group $G_l$ ($1 \leq l \leq L$) such that $c(i) = l$ if $i \in G_l$. Let $w_l = \{w_{kl}\}_{k=1}^p$ be a variable weighting for each individual group $G_l$. The goal is to find $c*$ and $\{w_l\}_l^L$ that minimize a criterion $Q(c, \{w_l\}_l^L)$ that measures the distance between groups and within each group. The authors suggest

$$Q(c, \{w_l\}_l^L) = \sum_{l=1}^L W_l \sum_{c(i)=l} \sum_{c(j)=l} D_{ij}(w_l),$$

$$D_{ij}(w_l) = \sum_{k=1}^p w_{kl} d_{ijk},$$

$$d_{ijk} = \delta_{ijk}/s_k,$$

$$\delta_{ijk} = |x_{ik} - x_{jk}|,$$

$$s_k = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \delta_{ijk}.$$ 

Thus, $Q(c, \{w_l\}_l^L)$ acts as a weighted average over all the groups of the within group mean distance between all pairs of objects assigned to the same group. The feature selection seeks to find an optimal weight $w$ as part of the clustering problem by jointly minimizing $Q(c, \{w_l\}_l^L)$. This method can be viewed as an enhancement to
distance-based clustering methods, where groups of objects can be associated with different and, possibly overlapping, subsets of relevant attributes.

Mixture models are frequently used in the second approach. Kim et al. (2006) proposed a method that selects the discriminating variables while fitting a $G$ components normal mixture model, where $G$ is unknown. The selection of discriminating variables is achieved by introducing a latent $p$-vector with binary entries, with 1s indicating discriminating variables. The non-discriminating variables are assumed to originate from a single, common distribution, while the discriminating variables follow the mixture model. Raftery and Dean (2006) recast the variable selection problem as a model comparison problem addressed using Bayes factors. In contrast to Kim et al. (2006), the non-discriminating variables are assumed to be conditionally independent of the cluster labels conditioned on the discriminating variables. Lee and Li (2012) introduced a ridgeline-based separability measure for variable selection under a mixture model. The separation between clusters is quantified using critical points on the ridgeline between two modes. A forward selection algorithm was used to select the variable subset that maximizes the total separability among all the clusters. In this method, the subset of relevant variables is the same across all the clusters. This constraint is clearly unrealistic and undesirable in many applications; it is one that we explicitly reject in developing more incisive discriminatory information measures for cluster-specific variable subset selection.
2

Bayesian EM and subgroup identification in Gaussian DP mixtures

2.1 Introduction

In this chapter, a new Bayesian Expectation-Maximization (EM) algorithm is developed for fitting DPM models. Various MCMC methods have been developed to obtain posterior inference in DPM models, such as Neal (2000). Because the mixture model cannot distinguish component labels, known as the label switching problem, problems arise when the posterior samples obtained from MCMC are analyzed. Given a mixture model with $J$ components, due to the invariance of the mixture distribution with respect to the permutation of the component labels, there are $J!$ symmetric regions in the posterior distributions, and the posterior is highly multi-modal. MCMC samplers typically switch unpredictably from mode to mode between the iterations. Hence, the usual ergodic averages are not appropriate for estimating component-specific quantities. Therefore, relabeling methods must be developed to extract meaningful summaries from posterior samples. On the other hand, the EM algorithm is extensively used to estimate finite mixture models. The
EM algorithm provides a set of maximum a posterior (MAP) estimates at one mode of the posterior distributions; thus, it avoids the need for relabeling, which defines only point estimates.

In conventional mixture model based clustering, each mixture component is treated as one cluster; e.g., it is often assumed that a cluster is normally distributed. Therefore, each Gaussian mixture component corresponds to one cluster in a Gaussian mixture models. In this chapter, a mode search algorithm is developed to aggregate Gaussian components to model non-Gaussian clusters such that one cluster is allowed to contain several mixture components.

The chapter is organized as follows. Section 2.2 provides a general introduction to the EM algorithm. Section 2.3 introduces the new Bayesian EM algorithm for fitting DPM models and defining non-Gaussian clusters from a Gaussian mixture. Section 2.4 presents the mode searching algorithm to group the components by mode. Section 2.5 provides a summary of the chapter.

2.2 Introduction to Bayesian EM algorithm

The EM algorithm (Dempster et al., 1977) was originally developed for finding maximum likelihood estimates from incomplete data in a frequentist framework. The algorithm can be applied equally well to finding modes of the posterior distribution by introducing latent variables in a Bayesian framework. Given a statistical model consisting of a set of observations $x$, a set of latent variables $z$, and a vector of parameters $\theta$, the EM algorithm iteratively computes MAP estimates between the E-step and the M-step.

More explicitly, the Bayesian EM algorithm iteratively finds a mode of $p(\theta|x)$, or equivalently, $\log p(\theta|x) = \log \int_z p(\theta, z|x)dz$. The fundamental idea of the EM algorithm is to maximize a lower bound on the log posterior, not the log posterior
directly. Let \( l(\theta) = \log \int_z p(\theta, z|x)d_z \), and \( q(z) \) as an arbitrary distribution of \( z \), then

\[
l(\theta) = \log \int_z p(\theta, z|x)d_z
\]

\[
= \log \int_z \frac{q(z)}{q(z)} p(\theta, z|x)d_z
\]

\[
= \log \int_z q(z) \frac{p(\theta, z|x)}{q(z)}d_z,
\]

According to Jensen’s inequality, which states that, for \( \lambda \in (0, 1) \) and \( \phi \) a concave function,

\[
\lambda \phi(x) + (1 - \lambda) \phi(y) \leq \phi(\lambda x + (1 - \lambda)y),
\]

or, in general, \( E[\phi(X)] \leq \phi(E[X]) \), we then have

\[
\log \int_z q(z) \frac{p(\theta, z|x)}{q(z)}d_z \geq \int_z q(z) \log \frac{p(\theta, z|x)}{q(z)}
\]

\[
= J(q, \theta).
\]

Further, let \( q(z) = p(z|\theta, x) \). Then

\[
\frac{p(\theta, z|x)}{q(z)} = \frac{p(\theta, z|x)}{p(z|\theta, x)} = \frac{p(\theta, z|x)}{p(\theta, z|x)/p(\theta|x)} = p(\theta|x)
\]

regardless of what \( z \) is. Therefore, with this selection of \( q \), \( J(q, \theta) = l(\theta) \) and the lower bound is achieved. In addition, because \( J(q, \theta) \leq l(\theta) \), this choice of \( q \) maximizes \( J \) for any fixed \( \theta \). Hence the EM algorithm functions by iteratively applying the following two steps under the current estimate of the parameters \( \theta^{(t)} \),

\textit{E step}: find \( q^{(t)} = \arg\max_q J(q, \theta^{(t)}) \), which is just \( p(z|x, \theta^{(t)}) \),

\textit{M-step}: find \( \theta^{(t)} = \arg\max_\theta J(q^{(t)}, \theta) \), which is just to maximize \( \int_z q(z) \log p(\theta, z|x) \), or the expected value of the log posterior function, with respect to the conditional distribution of \( z \) given \( x \) under the current estimate of the parameters \( \theta^{(t)} \), or the so-called “Q-function” denoted by \( Q(\theta|\theta^{(t)}) \).
The above algorithm can thus be re-expressed as:

**E-step:** To compute

\[ Q(\theta|\theta^{(t)}) = E[\log \{ p(\theta, z|x) \} | \theta^{(t)}, x] \]

**M-step:** Find the parameter that maximizes this quantity:

\[ \theta^{(t)} = \arg\max_{\theta} Q(\theta|\theta^{(t)}) \]

It is easy to see that after the first step, \( J(q^{(t)}, \theta^{(t)}) \geq J(q^{(t-1)}, \theta^{(t)}) \), and after the second step, \( J(q^{(t)}, \theta^{(t+1)}) \geq J(q^{(t)}, \theta^{(t)}) \). The end result of this is

\[ l(\theta^{(t+1)}) = J(q^{(t+1)}, \theta^{(t+1)}) \]
\[ \geq J(q^{(t)}, \theta^{(t+1)}) \]
\[ \geq J(q^{(t)}, \theta^{(t)}) = l(\theta^{(t)}) \]

Therefore, each EM iteration can only increase the posterior density, guaranteeing convergence to a local maximum. The EM algorithm should be run at multiple different starting points to potentially explore many posterior modal regions.

### 2.3 Bayesian EM algorithm in Dirichlet process mixtures

The new Bayesian EM algorithm developed in this chapter is for fitting the truncated DPM model, in which \( p \)-vector observations \( x \) have density

\[ g(x|\Theta) = \sum_{j=1}^{J} \pi_j N(\mu_j, \Sigma_j) \]  

(2.1)

with prior hierarchically defined as follows:

\[ \pi_1 = V_1, \quad \pi_j = (1 - V_1), \ldots, (1 - V_{j-1})V_j, \quad 1 < j < J, \]
\[ V_j \mid a \sim Be(1, a), \quad j = 1, \ldots, J - 1, \]
\[ a \sim Ga(e, f), \]
\[ \mu_j \mid \Sigma_j \sim N(m, t\Sigma_j), \]
\[ \Sigma_j \sim IW(k + 2, kK) \]
for specified hyperparameters \((e, f, m, t, k, K)\) and some fixed (large) upper bound \(J\) on the number of effective components. Based on observing the random sample \(x_{1:n} = \{x_1, \ldots, x_n\}\), the posterior for \(\Theta = \{\mu_{1:J}, \Sigma_{1:J}, V_{1:J-1}\}\) and latent variables \((a, z_{1:n})\) are of interest, where \(z_{1:n} = \{z_1, \ldots, z_n\}\) is the set of latent configuration indicators, viz. \(z_i = j\) if, and only if, \(x_i\) comes from normal component \(j\).

Numerical search to identify modes of the posterior \(p(\Theta|x_{1:n})\) via the new EM procedure for truncated Dirichlet mixture models is as follows. This extends the standard method treating the latent variables \((a, z_{1:n})\) as missing data, iterating over \(t = 0, 1, \ldots\), based on starting parameter values \(\Theta^{(0)}\). At iterate \(t + 1:\)

**E-step:** Define \(Q(\Theta|\Theta^{(t)}) = \mathbb{E}[\log p(\Theta, z_{1:n}, a|x_{1:n})| \Theta^{(t)}, x_{1:n}]\).

For given parameters \(\Theta\), denote the conditional posterior classification probabilities by \(\pi_{ij} = \pi_j(x_i) = \pi_j N(x_i | \mu_j, \Sigma_j) / g(x_i | \Theta)\) and define \(\hat{a} = \mathbb{E}[a|\Theta, x_{1:n}] = (J + e - 1)/(f - \sum_{j=1}^{J-1} \log(1 - V_j))\). Then \(Q(\Theta|\Theta^{(t)})\) is given, up to a constant, by

\[
Q(\Theta|\Theta^{(t)}) = c + \sum_{j=1}^{J} \left[ \sum_{i=1}^{n} \pi_{ij}^{(t)} \log \{ \pi_j N(x_i | \mu_j, \Sigma_j) \} + \log [p(\mu_j | \Sigma_j) p(\Sigma_j)] \right] \\
+ \sum_{j=1}^{J-1} (\hat{a}^{(t)} - 1) \log(1 - V_j).
\]

**M-step:** Compute \(\Theta^{(t+1)} = \arg \max_\Theta Q(\Theta|\Theta^{(t)})\). Letting \(c_j^{(t)} = \sum_{i=1}^{n} \pi_{ij}^{(t)}\), this yields the following, with index \(j\) running from \(j = 1, \ldots, J\) except as noted for the
\[ V_j^{(t+1)} = \min\{1, \ c_j^{(t)}/[\hat{a}^{(t)} - 1 + \sum_{r=j}^J c_r^{(t)}]\} ; \]

\[ \pi_1^{(t+1)} = V_1^{(t+1)}, \quad \pi_j^{(t+1)} = (1 - V_1^{(t+1)}) \cdots (1 - V_{j-1}^{(t+1)}) V_j^{(t+1)}, \quad j = 2, \ldots, J; \]

\[ \mu_j^{(t+1)} = (m + tc_j^{(t)} \bar{x}_j)/(1 + tc_j^{(t)}) \quad \text{where} \quad \bar{x}_j = \frac{\sum_{i=1}^n \pi_{ij}^{(t)} x_i}{c_j^{(t)}}; \]

\[ \Sigma_j^{(t+1)} = S_j^{(t)}/(c_j^{(t)} + k + 2p + 3) \quad \text{where} \]

\[ S_j^{(t)} = kK + c_j^{(t)}(\bar{x}_j - m)(\bar{x}_j - m)/(1 + tc_j^{(t)}) + \sum_{i=1}^n \pi_{ij}^{(t)} (x_i - \bar{x}_j)(x_i - \bar{x}_j). \]

A key practical point to note is that an identified posterior mode will typically identify fewer than the maximum specified number of components, so providing an automatic indicator of effective number of components from a mode search. This arises when the M-step optimization over the \( V_j \) yields \( V_j = 1 \) for \( j \geq J' \), for some \( J' < J \). The developed algorithm is used extensively in Section 3.3.

2.4 Non-Gaussian subgroups by aggregating Gaussian components

In many mixture model-based clustering analysis, it is often assumed that number of mixture components corresponds to the number of clusters in the data. However, this assumption may not always be true in practice. Svensen and Bishop (2004) tried to relax the normality assumption and developed a robust Bayesian mixture modeling based on Student-t distributions, which are heavier tailed than Gaussians and hence more robust to outliers. However, they still treated each component as one cluster. In this section, the mode searching algorithm is developed to allow one cluster to contain several components depending on whether they merge into one mode. Hence, relaxing the normality assumption, in addition, thus allows the mixture model to be
more flexible in modeling data coming from populations consisting of non-Gaussian subgroups. The need for relaxing both of the assumptions are exemplified by the analysis of real flow cytometry data in both Chapter 3 and 4.

The basic concept here is that a Gaussian mixture with many components can flexibly represent a set of clusters, with each cluster - or subgroup - defined by the sub-mixture of a subset of Gaussians. When a few Gaussian components are “close”, they can be identified as plausibly representing a subgroup. One key and natural idea of “closeness” here is to (a) find modes in the overall mixture, then (b) identify two, or more, Gaussian components as associated with a particular mode via a “basin of attraction” argument. This will then agglomerate Gaussian components around each mode and so define non-Gaussian subgroups.

Given a set of parameters, whether posterior mode estimates or a sample from the posterior, for the Gaussian mixture of equation (2.1), I follow previous work (Chan et al., 2008; Finak et al., 2009) in defining clusters by aggregating proximate Gaussian components. That is, identify $C \leq J$ clusters with index sets $I_c$ containing components indices $j$ for each subtype $c = 1 : C$. Then

$$\alpha_c = \sum_{j \in I_c} \pi_j$$

and

$$g(x) = \sum_{c=1}^{C} \alpha_c f_c(x)$$

where

$$f_c(x) = \sum_{j \in I_c} (\pi_j / \alpha_c) N(x | \mu_j, \Sigma_j), \ c = 1, \ldots, C.$$  

Grouping components into clusters can be done by associating each of the Gaussian components with the closest mode of $g(x)$. By running an efficient modal search beginning at each of the $\mu_j$ we can swiftly identify the set of modes in $g(x)$ together with the indicators of which mode each Gaussian component is attracted too. The number of modes so identified is $C$, taken as the realized number of clusters in the mixture.

Efficient numerical optimization uses the mode trace function for Gaussian mixtures. Define precision matrices $\Omega_j = \Sigma_j^{-1}$. Mode search start with iteration index
\[ x^{i+1} = A(x^i)^{-1} \sum_{j=1}^{J} \gamma_j(x^i) \Omega_j \mu_j \]

where \( A(x) = \sum_{j=1}^{J} \gamma_j(x) \Omega_j \) and \( \gamma_j(x) = \pi_j \mathcal{N}(x|\mu_j, \Sigma_j) \). This is a convergent local mode search that is broadly useful to quickly identify modes, antimodes and ridge lines between them in the contours of Gaussian mixtures, and typically takes just a few iterates. A second derivative of \( g(x) \) evaluated at any identified stationary point then identifies it as a mode or antimode. Rather than being interested in all modes of \( g(x) \), I am here only interested in those that define basins of attraction for the mixture components in order to find the sets \( I_c \) of component indicators related to different modes. Hence this numerical search is run \( J \) times, initializing at \( x^0 = \mu_j, j = 1 : J \) in turn, and the unique modes are recorded so identified as well as the sets \( I_c \) of Gaussian components attracted to each in this search.

### 2.5 Discussion

In this chapter, I have developed a new Bayesian EM algorithm in fitting a DPM model. Due to the multi-modal nature of the posterior distributions for the mixture model, the parameter estimates from the EM algorithm avoid the label switching problem, hence can be used to make posterior inferences directly, albeit only in terms of posterior point estimate. As described in Section 2.3, the EM algorithm, under each iteration, involves a computation of conditional posterior classification probabilities \( \pi_j(x_i) \) for every observation \( i, i = 1 : n \), and every mixture component \( j, j = 1 : J \). This posterior probabilities computation imposes a tremendous computational burden when fitting the DPM models with increasing heterogenous and/or increasing large data sets. Suchard et al. (2010) exploited a distributed GPU implementation on the developed Bayesian EM algorithm. Thus, the algorithm can
be applied easily and routinely across analysis using the GPU-based parallel implementa-
tion.

Under the clustering problem, both the number of clusters and cluster distributions are always unknown, creating difficulties in the use of mixture models for density estimation and clustering. In this chapter, the two problems are solved by defining a cluster distribution as a set of proximate Gaussian components, permitting the model to flexibly accommodate different types of clusters. The idea of merging mixture components serves as a fundamental concept in performing cluster analysis under mixture models throughout the entire dissertation.
Discriminative information analysis in mixture modeling

3.1 Introduction

In this chapter, I am interested in the general question of identifying subsets of variables that play roles in discrimination of subpopulations (also referred to as subgroups or clusters) in the context of multivariate mixture modeling. The aim is to define an effective, computationally accessible approach to variable subset assessment and prioritization with regard to the ability of each subset to discriminate one or more subpopulations from the rest. A good example of the need for such a method is in routinely applied biological cell assays using flow cytometry that generate multiple, large data sets in 10-20 dimensions (e.g. Chan et al., 2008; Finak et al., 2009); automated mixture model fitting and subsequent discriminatory analysis to prioritize subsets of variables will have a major impact on advancing statistical work in these and other areas.

Suppose $x_{1:n} = \{x_1, ..., x_n\}$ is a random sample from a $p$—dimensional, $C$—component
mixture distribution with density function

\[ g(x) = g(x|\Theta) = \sum_{c=1}^{C} \alpha_c f_c(x|\theta_c) \]  

(3.1)

where each subpopulation density \( f_c \) has its own parameters \( \theta_c \) and component probability \( \alpha_c \), \( (c = 1, \ldots, C) \), and \( \Theta = \{C, \alpha_{1:C}, \theta_{1:C}\} \) is the full set of parameters. Based on fitting the model to observed data \( x_{1:n} \), the following question are addressed. For each component \( c \),

1. Which subsets of the \( p \) variables, if any, contribute in meaningful ways to discrimination of \( f_c \) from the other components?

2. Are there variables that are irrelevant to discrimination of \( f_c \)?

3. Are there single or small subsets of variables that characterize \( f_c \) alone and play no roles in discriminating other components?

4. Can we rank subsets of variables by their discriminatory ability with respect to \( f_c \)?

The general questions relate to variable selection in mixtures, earlier studied by, for example, Raftery and Dean (2006) and Kim et al. (2006). In contrast to these approaches, I address the discriminatory analysis following model fitting: the model is first assumed available with either plug-in parameter estimates or Markov chain Monte Carlo based posterior samples (e.g. Lavine and West, 1992; West, 1992, 1997; Richardson and Green, 1997) and I then aim to interrogate the model to define variable subset discriminatory measures. Further, the issue that very different subsets of variables may play roles in differentiating different mixture components is directly addressed. Also, I am keen to define computationally effective approaches so as to enable access and routine use, and an ability to scale to larger models.
I then provide theoretical development and examples in Bayesian mixture models using standard truncated Dirichlet process mixtures (e.g. Escobar and West, 1995; MacEachern and Müller, 1998; Ishwaran and James, 2001; Müller and Quintana, 2004), as it offers an approach to handling uncertainty about $C$ subject to a specified large upper bound.

The examples and much applied interest lie in mixture models where each component density $f_c$ may have a non-Gaussian form. One effective approach is to fit an encompassing mixture of Gaussians and then aggregate subsets of the fitted densities; that is, each $f_c$ is itself represented as a mixture of, typically, a small number of Gaussians (e.g. Chan et al., 2008; Finak et al., 2009), as I developed in Chapter 2. Section 3.3 uses Bayesian computational methods for Gaussian mixtures and the subsequent construction of non-Gaussian subpopulation densities. The computational work in this chapter also utilizes efficient parallel implementations of Markov chain Monte Carlo for large-scale mixtures (Suchard et al., 2010). The model fitting via Markov chain Monte Carlo also uses an effective component relabeling approach (Cron and West, 2011).

3.2 Density concordance and discriminative information measure

3.2.1 Definitions

The concordance between two density functions $f(x), h(x)$ is $\delta = \int f(x)h(x)dx$, a natural measure of agreement or overlap of the densities; $\delta$ takes its maximum value when the densities agree exactly and otherwise decays towards zero as the densities become more separated. Concordance was discussed as the basis of a similarity distance between densities by Scott and Szewczyk (2001). It can be seen that $\delta = \mathbb{E}[f(x)]$ where the expectation is over $x \sim h(\cdot)$. In assessing how different component $f_c$ is to the set of remaining components of the mixture in equation (3.1), it is natural to look at the concordance between $f_c$ and the renormalized mixture defined by the
remaining \( C - 1 \) components. This motivates the following.

Refer to the mixture in equation (3.1). For notational clarity, here I write \( f_c(x) = f_c(x|\theta_c) \), the dependence on parameters being implicit. Also I use \( E_c[\cdot] \) for expectation under \( x \sim f_c(\cdot) \). The discriminative information measure for mixture component \( c \) is defined as 
\[
\delta_c = \frac{\Delta_c}{\delta_c} \quad \text{where:}
\]
\[
\delta_c = (1 - \alpha_c) E[f_c(x)|x \sim f_c(\cdot)] = \sum_{e=1, e \neq c}^C \alpha_e \delta_{ec},
\]
\[
\Delta_c = E[f_c(x)|x \sim g(\cdot)] = \sum_{e=1}^C \alpha_e \delta_{ce},
\]
\[
\delta_{ce} = E_c[f_c(x)], \quad e = 1, \ldots, C.
\]
So \( d_c \) is the expected value of \( f_c(x) \) for observations that are not from subpopulation \( c \) relative to its global expected value; it measures discrimination of component \( c \) from the rest based on concordance. It can be seen that \( \Delta_c = \delta_c + \alpha_c \delta_{cc} \) which, together with \( \alpha_c \delta_{cc} \geq 0 \), implies that \( 0 < d_c \leq 1 \) so discrimination is measured on a standardized scale. A low value of \( d_c \) represents a high level of discrimination of component \( c \) from the rest implied by good separation between \( f_c \) and the other densities, while a high value represents poor discrimination. Discrimination is thus on a \( 0 - 1 \) scale with \( 100(1 - d_c) \) being the percent discrimination obtained.

In comparing discrimination based on different subsets of variables, I make explicit in the notation which variables are used. For any subset of variables \( h \subseteq \{1 : p\} \), denote by \( d_c(h) \) the discriminative information measure when restricting to only those variables. It can be interpreted that subset \( h \) provides \( 100(1 - d_c(h)) \% \) discrimination for mixture component \( c \).

Given a specified mixture on the full set of \( p \)-variables in \( x \), the computation of \( d_c(h) \) for each component \( c = 1 : C \) is done by directly marginalizing to the variable subset \( h \). This way the issue of evaluating discriminatory subsets can be carried out.
for all components and all subsets of variables based on the fitted model, without refitting. By exploring subsets \( h \), I can then automatically generate ranked sets of variables for each component and address the questions above. Notice that, at an extreme, if one or a subset of variables \( h \) is independent of the rest and has the same distribution over all components, then \( d_c \) will take the same value when computed in the mixture model analysis with or without those variables, showing their irrelevance.

### 3.2.2 Theoretical insights

The problem of assessing discriminatory relevance of subsets of variables is related to the question of cluster separability. Dy and Brodley (2004) studied and utilized the scatter separability criterion to measure the quality of clustering and perform feature selection. \( \Delta_c \) can be related to the sum of within-cluster separability and between-cluster separability with respect to cluster \( c \). From the definition of \( \Delta_c = \delta_c + \alpha_c \delta_{cc} \), \( \alpha_c \delta_{cc} \) resembles within-cluster separability in the sense that it reaches its maximum if \( f_c \) is a Dirac measure; it decreases as \( f_c \) has flatter tails, so larger \( \alpha_c \delta_{cc} \) means smaller within-cluster separability. \( \delta_c \) resembles between-cluster separability, i.e. it measures how scattered the cluster \( c \) is from the remaining clusters. The value decreases as cluster \( c \) becomes more separated from the rest, so smaller \( \delta_c \) means larger between-cluster separability. Hence, larger \( \alpha_c \delta_{cc} \) and smaller \( \delta_c \) are preferred for “good” discrimination. In addition, \( d_c \) can be interpreted as the proportion of between-cluster separability. This interpretation could be helpful in determining the number of variables to be selected, as discussed in detail in Section 3.2.4.

It can be shown that \( d_c \) is biased with respect to dimension, i.e. \( d_c \) tends to select subsets with higher dimensions under certain conditions. This bias can be illustrated by the following simple example. Assume the mixture model

\[
g(x) = \alpha_1 f_1(x) + (1 - \alpha_1) f_2(x) = \alpha_1 N(\mu_1, \Sigma_1) + (1 - \alpha_1) N(\mu_2, \Sigma_2),
\]

where each Gaussian density function describes one cluster. Suppose \( h_1 \) and \( h_2 \) are different subsets of
variables with sizes \( p_1 \) and \( p_1 + 1 \) respectively, \( h_1 \subset h_2 \), s.t. \( h_1 = h_2(1 : p_1) \). Further assume that \( h_1 \) and \( h_2(p_2) \) are uncorrelated within each cluster. Suppose we are interested in cluster 1, then
\[
d_1(h_1)^{-1} = 1 + \alpha_1 \delta_{11}/(1 - \alpha_1)\delta_{12} = 1 + \nabla_1(h_1), \quad \nabla_1(h_1) = \alpha_1 \delta_{11}/(1 - \alpha_1)\delta_{12},
\]
\[
d_1(h_2)^{-1} = 1 + \nabla_1(h_1)(2\sigma_{1p_2}^2)^{-0.5}/(\sigma_{1p_2}^2 + \sigma_{2p_2}^2)^{-0.5} \exp\{-((\mu_{1p_2} - \mu_{2p_2})^2/(2(\sigma_{1p_2}^2 + \sigma_{2p_2}^2))\}.
\]
It can be easily shown that \( d_1(h_1) = d_1(h_2) \) when \( \mu_{1p_2} = \mu_{2p_2} \) and \( \sigma_{1p_2} = \sigma_{2p_2} \), i.e. \( d_1 \) will assign equal values to both subsets if the additional variable is non-informative in that it contains no additional information in differentiating cluster 1 from the other. In this situation, a smaller subset is preferred. If \( \mu_{1p_2} = \mu_{2p_2} \), but \( \sigma_{1p_2} < (>\sigma_{2p_2} \), then \( d_1(h_1) > (\leq)d_1(h_2) \). Thus, by comparing relative variability among clusters, if the cluster of interest has smaller variance in the additional dimension, \( h_2 \) is preferred, as smaller \( \sigma_{1p_2} \) results in relatively smaller within-cluster separability on the \( p_2 \) dimension; otherwise, \( h_1 \) is preferred as \( h_2(p_2) \) will only contribute noise.

If \( \mu_{1p_2} \neq \mu_{2p_2} \), but \( \sigma_{1p_2} = \sigma_{2p_2} \), \( d_1(h_2) < d_1(h_1) \) always, i.e. \( d_1 \) will increase strictly with every added variable. In this situation, some threshold should be set on the decrease of \( d_1 \) for variable selection, as discussed in Section 3.2.4.

In the general setting, suppose the important variables are denoted by \( x \), where additional variable(s) to be considered are denoted by \( y \). Assume that \( y \) is non-informative and \( f_e(x, y) = f_e(x)g(y) \) for \( e = 1, \ldots, C \). Then:
\[
d_c(x)^{-1} = 1 + \nabla_c(x), \quad \nabla_c(x) = \alpha_c \delta_{ce}(x)/\delta_c(x),
\]
\[
d_c(x, y)^{-1} = 1 + \nabla_c(x) \int f_c^2(y)dy/\int f_c(y)f_{-c}(y)dy.
\]
We see that \( d_c(x, y) \leq d_c(x) \) if and only if \( \int f_c^2(y)dy \geq \int f_c(y)f_{-c}(y)dy \), which indicates the within-cluster separability is relatively smaller than the between-cluster separability. Clearly \( d_c(x, y) \geq d_c(x) \) otherwise. This implies that when non-informative variables only mask the clustering, then \( d_c \) reaches a minimum; otherwise, \( d_c \) will de-
crease as variables are added. Comparing the scatter separability criterion in Dy and Brodley (2004), which increases monotonically with dimension, our defined measure can outperform when ∫ f_c(y)dy < ∫ f_c(y)\overline{f}_c(y)dy. This can occur, e.g., under the above mixture of two normal components case, when all the clusters have the same locations, but the cluster of interest has larger variance under the added dimension.

Lee and Li (2012) introduced a ridgeline-based separability measure for variable selection. Their measure has an explicit expression only under a two-component normal mixture with identity covariance matrices. Consider a mixture density

\[ 0.5N(x|\mu_1, I) + 0.5N(x|\mu_2, I), x \in \mathbb{R}^p \]

Let \( \beta = ||\mu_1 - \mu_2||^2 / 2 \). Then their separability measure between the two normal densities becomes \( S(\beta) = 1 - 2e^{-0.25\beta} / (1 + e^{-\beta}) \) for \( \beta > \beta^* \), otherwise \( S(\beta) = 0 \), where \( \beta^* \in (2.4, 2.5) \). The measure \( d \) becomes \( d_\beta = d_{1,\beta} = d_{2,\beta} = e^{-0.5\beta} / (1 + e^{-0.5\beta}) \). The measures \( S(\beta) \) and \( 1 - d_\beta = 1 / (1 + e^{-0.5\beta}) \) are compared in Figure 3.1; each is monotonically increasing for \( \beta \) greater than 2.5. In addition, \( 1 - d_\beta \) is consistently
above $S(\beta)$ suggesting that the two measures are consistent in terms of preserving the order of the important subsets, but differ on the scale. Each increases strictly with every added variable; therefore, in considering variable selection, it is better to threshold the rate of change. Figure 3.2 shows the rates of change of the two measures. The black curve is consistently above the dashed line showing that $d_\beta$ is relatively robust to the small change of $\beta$ when the two means are separated. In addition, as the rate of change for $d$ is relatively small, it may be better to threshold the rate of change on the log scale. The threshold value is chosen based on the reference model $0.5N(x|0, I) + 0.5N(x|\mu_2, I), x \in R$, and $\beta = 7$. The rate of change for $\log(d_\epsilon)$ at $\beta = 7$ is 0.0147, therefore, the threshold value is chosen to be 0.015.

3.2.3 Misclassification rate connection

The problem of assessing discriminatory relevant of subsets of variables is related to the question of classification accuracy, though raises different issues that the new
discriminatory measure addresses. I am not interested, per se, in misclassification rates as they involve the base rates \( \alpha_c \) that do not factor into consideration of how subsets of variables are relatively distributed within subpopulations. However, it is of interest to relate \( d_c \) to misclassification rates for further interpretation. As I now show, it also turns out that a simple approximation yields good estimates of expected misclassification rates based on the concordance measures and some associated, also easily computable quantities. This is also useful since estimating misclassification rates is a computational challenge that typically requires additional simulation-based computations and this is not attractive for routine analysis, particularly with higher dimensional models.

Write \( \alpha_c(x) = \alpha_c f_c(x)/g(x) \) for the component \( c \) classification probability at a point \( x \) and \( r_c = E[\alpha_c(x)|x \neq f_c(\cdot)] \) for the expected misclassification rate for component \( c \). From the definitions of \( r_c \) and \( d_c \) we can deduce the expressions \( (1 - \alpha_c)r_c/\alpha_c = 1 - \alpha_c E_c[f_c(x)/g(x)] \) and \( (1 - \alpha_c)d_c = 1 - \alpha_c E_c[f_c(x)/E_c[g(x)] \). Write \( V_c[\cdot] \) and \( C_c[\cdot] \) for variance and covariance under \( x \sim f_c(x) \), respectively. A direct, second-order Taylor series approximation of the integrand of \( E_c[f_c(x)/g(x)] \) coupled with the corresponding first-order approximation of the variance \( V_c[f_c(x)/g(x)] \) then yields the relationship

\[
    r_c = \hat{r}_c + \epsilon_c, \quad \hat{r}_c = \alpha_c d_c - \{1 - (1 - \alpha_c)d_c\} k_c \alpha_c/(1 - \alpha_c)
\]

where \( \epsilon_c \) is a zero-mean approximation error with variance \( \tau_c^2 \); the terms \( k_c, s_c \) and \( \tau_c \) are defined by

\[
    k_c = \frac{V_c[g(x)]}{E_c[g(x)]^2} - \frac{C_c[f_c(x), g(x)]}{E[f_c(x)]E_c[g(x)]}
\]

\[
    s_c = \frac{V_c[f_c(x)]}{E_c[f_c(x)]^2} + \frac{V_c[g(x)]}{E_c[g(x)]^2} \quad \text{and} \quad \tau_c^2 = \frac{\alpha_c^4 E_c[f_c(x)]^2}{(1 - \alpha_c)^2 E_c[g(x)]^2} \left\{s_c - \frac{2 C_c[f_c(x), g(x)]}{E[f_c(x)]E_c[g(x)]}\right\}.
\]

These are easily computable in Gaussian and other mixtures. As a result, the quantity \( \hat{r}_c \) is an easily computable approximation to \( r_c \) with corresponding uncertainty.
assessment defined by $\tau_c$. Examples in Section 3.3 show that the lead term $r_c \approx \alpha_c d_c$ in equation (3.2) is dominant and often provides a useful approximation alone, ignoring the second term; it also directly links discriminatory evidence to misclassification rates.

### 3.2.4 Variable selection procedures

In this chapter, I consider the situations that the model in the full space is satisfactory, that the variable selection procedures are based on the parameters estimates from the full model. This assumption can be easily relaxed by performing additional model fitting step for each examined subset of variables. The Bayesian EM algorithm detailed in Section 2.3 for model fitting can be run very fast by utilizing GPU programming. Under the ideal case when the number of variables to be considered is moderate and $d_c$ is not biased towards dimensionality, an exhaustive search of all possible subsets of variables can be performed, and the optimal variable subset gives the lowest $d_c$. Otherwise, a forward selection algorithm can be used for variable selection: Let the set of variables already been selected after $k$ iterations of the forward selection be $S_k$, for $k = 1, ..., p$. Under the algorithm, $S_k$ is a nested sequence. Denote the variables not included in $S_k$ by $S_{-k} = x_{1:p} \setminus S_k$. Thus at the $(k + 1)th$ iteration,

1. For each $x_j \in S_{-k}$, compute $d_c(S_k \cup x_j)$

2. Choose $x_j^*$ such that $x_j^* = \arg\min_{x_j \in S_{-k}} d_c(S_k \cup x_j)$

3. Change $S_k = S_k \cup x_j^*$; $S_{-k} = S_{-k} \setminus x_j^*$

4. Stop if the percentage decrease in $\log(d_c)$ by adding the selected variable is below a threshold $\epsilon$. Otherwise, change $k + 1$ to $k$, and repeat the above steps.
3.3 Simulated data examples

The truncated DPM model is employed for simulation studies using the Bayesian EM for fitting posterior models. Recall this has the desirable property of automatically cutting-back to mixtures with fewer components than a specified upper bound, so providing an automatic indicator of effective number of components from a mode search. Follow-on posterior simulation then refines these initial posterior summaries using MCMC.

Example 1. A sample of size \( n = 5,000 \) was drawn from a \( p = 3 \)--dimensional mixture of 2 normal distributions in which only the first 2 variables carry discriminatory information; see Figure 3.3. Analysis allows up to 9 components using default, relatively vague priors. The Bayesian EM algorithm was run repeatedly from many random starting points; the highest posterior mode identified the correct 2 components. Using parameters set at this mode leads to the discriminative summaries in Table 3.1.

Table 3.1: Example 1. Discriminative measures \( d_c(h) \), Monte Carlo estimates of misclassification rates \( r_c(h) \) and the estimates and estimation standard error from equation (3.2). Calculations are based on the 3-dimensional Gaussian mixture with parameters estimated as posterior modes.

<table>
<thead>
<tr>
<th>( h )</th>
<th>1,2,3</th>
<th>1,2</th>
<th>1,3</th>
<th>2,3</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_1(h) )</td>
<td>( 2.65 \times 10^{-7} )</td>
<td>( 7.15 \times 10^{-7} )</td>
<td>0.1195</td>
<td>0.2638</td>
<td>0.2470</td>
<td>0.4681</td>
<td>0.2838</td>
</tr>
<tr>
<td>( r_1(h) )</td>
<td>( 3.1571 \times 10^{-5} )</td>
<td>( 5.4670 \times 10^{-5} )</td>
<td>0.1143</td>
<td>0.2124</td>
<td>0.1745</td>
<td>0.3651</td>
<td>0.2768</td>
</tr>
<tr>
<td>( \hat{r}_1(h) )</td>
<td>( 6.1894 \times 10^{-7} )</td>
<td>( 1.3844 \times 10^{-6} )</td>
<td>0.1158</td>
<td>0.2173</td>
<td>0.1506</td>
<td>0.3928</td>
<td>0.2906</td>
</tr>
<tr>
<td>( \tau_1(h) )</td>
<td>( 2.8756 \times 10^{-4} )</td>
<td>( 5.9775 \times 10^{-4} )</td>
<td>0.2900</td>
<td>0.3074</td>
<td>0.4100</td>
<td>0.2538</td>
<td>0.1871</td>
</tr>
<tr>
<td>( d_2(h) )</td>
<td>( 1.28 \times 10^{-7} )</td>
<td>( 2.51 \times 10^{-7} )</td>
<td>0.1038</td>
<td>0.2356</td>
<td>0.1691</td>
<td>0.3548</td>
<td>0.3697</td>
</tr>
<tr>
<td>( r_2(h) )</td>
<td>( 3.3139 \times 10^{-5} )</td>
<td>( 5.0979 \times 10^{-5} )</td>
<td>0.1101</td>
<td>0.2049</td>
<td>0.1682</td>
<td>0.3521</td>
<td>0.2669</td>
</tr>
<tr>
<td>( \hat{r}_2(h) )</td>
<td>( 2.9691 \times 10^{-7} )</td>
<td>( 5.2606 \times 10^{-7} )</td>
<td>0.1156</td>
<td>0.2126</td>
<td>0.1738</td>
<td>0.3498</td>
<td>0.2776</td>
</tr>
<tr>
<td>( \tau_2(h) )</td>
<td>( 3.2770 \times 10^{-6} )</td>
<td>( 4.1562 \times 10^{-6} )</td>
<td>0.1749</td>
<td>0.2598</td>
<td>0.1421</td>
<td>0.1140</td>
<td>0.2958</td>
</tr>
</tbody>
</table>

This clearly shows the \( d_c(h) \) measures correctly identify the first 2 variables as highly discriminatory and that the 3rd variable is redundant. Discrimination 100(1–
Figure 3.3: Pairwise scatter plots of a randomly selected subset of the \( n = 5,000 \) observations in Example 1. Dimensions 1 and 2 together discriminate the 2 normal components while dimension 3 is redundant.

\( d_c(h) \) is close to 100\% for \( h = (1, 2) \) and less than 80\% for other subsets of just 1 or 2 variables; adding variable 3 to \((1, 2)\) makes no practical change.

Using the standard Markov chain Monte Carlo method for posterior simulation, a posterior sample of size 20,000 was generated, initialized at the posterior mode used above. Computing the \( d_c(h) \) measures for each of these parameter draws gives an approximate posterior distribution for the discriminative evidence measures. Figure 3.4 shows some examples that also indicate that the plug-in values based on the posterior modal parameters are close to posterior modes for the \( d_c(h) \) measures.

Example 2. A second sample of size \( n = 5,000 \) was drawn from a \( p = 3 \)–dimensional mixture of 3 normal distributions with proportions \((0.05, 0.6, 0.35)\) and again with
Figure 3.4: Posteriors for $d_1(1,2,3)$, left, and $d_1(1,2)$, right, in Example 1. The posteriors are truncated at higher values for clarity. Dotted lines indicate posterior means. The $d_c(h)$ estimates based on mixture model parameters set at posterior modes are indicated by the dark dashed vertical lines.

the first 2 variables containing the primary discriminatory information. The 3rd variable masks the data structure when viewed in 1 or 2 dimensions, especially relative to the first, lower probability component; see Figure 3.5. The ability of the data to discriminate this low probability component is of interest here.

In an analysis again allowing up to 9 mixture components using default, relatively vague priors, the posterior mode search finds a highest mode at the correct 3 components; at the estimated parameters the discriminatory information summaries are as in Table 3.2.

This clearly shows $d_c(h)$ correctly identify the first 2 variables as highly discriminatory for component 1 and that the 3rd variable is redundant. Note that in this example the misclassification rates for component 1 are small even with non-discriminatory variable included, due to the low probability of component 1. Again a full posterior simulation defines posteriors for the $d_c$ measures that indicate the plug-
Figure 3.5: Pairwise scatter plots of a randomly selected subset of the \( n = 5,000 \) observations in Example 2.

Table 3.2: Example 2. Discriminative measures \( d_c(h) \), Monte Carlo estimates of misclassification rates \( r_c(h) \) and the estimates and estimation standard error from equation (3.2). Calculations are based on the 3-dimensional Gaussian mixture with parameters estimated as posterior modes.

<table>
<thead>
<tr>
<th>( h )</th>
<th>1,2,3</th>
<th>1,2</th>
<th>1,3</th>
<th>2,3</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_1(h) )</td>
<td>0.0002</td>
<td>0.0014</td>
<td>0.7937</td>
<td>0.5880</td>
<td>0.7933</td>
<td>0.7566</td>
<td>0.9422</td>
</tr>
<tr>
<td>( r_1(h) )</td>
<td>( 1.3340 \times 10^{-4} )</td>
<td>( 5.2623 \times 10^{-4} )</td>
<td>0.0286</td>
<td>0.0231</td>
<td>0.0339</td>
<td>0.0334</td>
<td>0.0448</td>
</tr>
<tr>
<td>( \dot{r}_1(h) )</td>
<td>( 2.2183 \times 10^{-5} )</td>
<td>( 1.3754 \times 10^{-4} )</td>
<td>0.0297</td>
<td>0.0215</td>
<td>0.0358</td>
<td>0.0357</td>
<td>0.0490</td>
</tr>
<tr>
<td>( \tau_1(h) )</td>
<td>( 1.0800 \times 10^{-4} )</td>
<td>( 9.3266 \times 10^{-4} )</td>
<td>0.0130</td>
<td>0.0178</td>
<td>0.0098</td>
<td>0.0093</td>
<td>0.0019</td>
</tr>
</tbody>
</table>
in estimated values reported lie squarely within the mass and close to the posterior mode.

Next I conduct experiments on three simulated data sets, comparing the developed discriminative measure with the ridgeline-based separability measure (RSM) introduced in Lee and Li (2012), as the scatter separability criterion (SSC) introduced in Dy and Brodley (2004) was already compared with RSM, and RSM was shown to be superior. My method will give different subsets of variables while RSM selects the same subset of variables for different cluster. The forward selection algorithm under both methods is based on the parameters estimates under the full model. The algorithm will be stopped if the rate of decrease in $\log(d_c)$ is below 0.015, and it will be stopped for RSM if the increase in RSM is lower than 0.01.

Example 3. Following the simulated data in Lee and Li (2012), a data set of size 6,000 and dimension $p = 8$ was generated. The first two dimension are generated according to

$$1/3N((3,9)^T, I) + 1/3N((5,6)^T, I) + 1/3\text{Unif}([0,8] \times [4,12]);$$

the uniform distribution serves to weaken the separation between the two normal components. The other six dimensions are non-informative, independent standard normal distributions. Figure 3.6 shows the scatter plot for a randomly selected subset of the generated sample.

The Bayesian EM algorithm is applied using up to 32 components using default, relatively vague priors. After aggregating some Gaussian components according to Section 2.4, the model identified 2 out of 3 effective modes, where the last mode corresponds to the uniform noise. The results of discriminative information measure were compared with RSM using forward selection algorithm, based on modal parameter estimates obtained in the full model. The comparisons are shown in Table 3.3. Similar to $RSM$, $d_c$ added the first two important variables sequentially. By the
Figure 3.6: Pairwise scatter plots of a randomly selected subset of the $n = 6,000$ observations in Example 3.

stopping criterion, both RSM and $d_c$ select the first two variables.

Example 4. Again following the simulated data in Lee and Li (2012), a data set of size 6,000 and dimension $p = 8$ was generated; the first two dimensions are generated according to a mixture of four Gaussian components with proportions 0.4, 0.2, 0.2 and 0.2, means $(6, 4)^T, (7, 10)^T, (2, 6)^T, (2, 12)^T$, and covariance matrices $1.5I_{2\times2}, 2I_{2\times2}, 1.5I_{2\times2},$ and $1.5I_{2\times2}$. The third and fourth dimensions are generated independently according to a two component Gaussian mixture with proportions 2/3 and 1/3, means $(6, 11)^T$ and $(5, 3)^T$, and a common covariance matrix with variances.
Table 3.3: Example 3. Discriminative measures \(d_c(h)\), for \(c = 1, 2\). Calculations for \(d_c(h)\) are based on the 8-dimensional Gaussian mixture with parameters estimated as posterior modes. The variable added at each step is listed with the corresponding values of \(d_c(h)\) and RSM. The last variable to be included under the pre-specified stopping rules is underlined.

<table>
<thead>
<tr>
<th>Step</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_1)</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>4</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>(d_1(h))</td>
<td>0.1940</td>
<td>0.1314</td>
<td>0.1311</td>
<td>0.1306</td>
<td>0.1300</td>
<td>0.1236</td>
<td>0.1059</td>
<td>0.0628</td>
</tr>
<tr>
<td>(x_1)</td>
<td>2</td>
<td>1</td>
<td>8</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>(d_2(h))</td>
<td>0.2934</td>
<td>0.1674</td>
<td>0.1645</td>
<td>0.1597</td>
<td>0.1525</td>
<td>0.1326</td>
<td>0.0909</td>
<td>0.0415</td>
</tr>
<tr>
<td>(x_1)</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>4</td>
<td>6</td>
<td>5</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>RSM</td>
<td>0.1260</td>
<td>0.2047</td>
<td>0.2050</td>
<td>0.2051</td>
<td>0.2052</td>
<td>0.2052</td>
<td>0.2052</td>
<td>0.2567</td>
</tr>
</tbody>
</table>

1, 2 and covariance 1. The rest of the dimensions are independent standard normals. Figure 3.7 shows a scatter plot of a randomly selected subset of the generated sample.

The Bayesian EM algorithm was used allowing up to 32 components using default, relatively vague priors. After aggregating Gaussian components according to Section 2.4, the model identified 4 effective clusters. The comparisons are shown in Table 3.4. According to the stopping criterion, RSM selects \(x_1, x_2,\) and \(x_4\). The \(x_2\) and \(x_4\) appear to be also important under \(d_1\) in four clusters, though \(d_1\) includes additional \(x_1\) and \(x_7\), \(d_2\) includes additional \(x_1\), and \(d_4\) includes additional \(x_3\). By looking at Figure 3.7, four clusters can be easily identified on the pairwise scatter plot of \(x_2\) vs. \(x_4\), thus graphically, both \(x_2\) and \(x_4\) should be selected. The inclusion of other variables is due to applying the same stopping criterion for all the clusters. E.g., for \(d_4(h)\), 0.0247 and 0.0222 are already very close, hence, the inclusion of \(x_3\) will not provide significant information. This suggests that different stopping criteria might be needed for different clusters.

Example 5. In the last example, a data set of size 6,000 and dimension \(p = 8\) was generated according to a four component Gaussian mixtures with proportions 0.3, 0.3, 0.3 and 0.1, where the last component is just background noise. The
Figure 3.7: Pairwise scatter plots of a randomly selected subset of the \( n = 6,000 \) observations in Example 4.

Mean vectors are \( (7, 0, 0, 0, 0, 0, 0, 5)^T \), \( (5, 5, 5, 0, 0, 0, 0, 0)^T \), \( (0, 0, 0, 5, 5, 0, 0, 0)^T \), \( 0_{8 	imes 1}^T \) and the covariance matrices for the first Gaussian component is diagonal matrix with diagonal elements \( (2, 2, 2, 2, 2, 2, 2, 0.5)^T \), the covariance matrices in the non-zero mean dimensions of the second and third components are

\[
\begin{pmatrix}
1.5 & 0.6 & 0.9 \\
0.6 & 1 & 0.3 \\
0.9 & 0.3 & 0.8
\end{pmatrix},
\begin{pmatrix}
1 & 0.1 & 0.9 \\
0.1 & 1.5 & 0.3 \\
0.9 & 0.3 & 2
\end{pmatrix},
\]

and the remaining dimensions are independent with variance 2. The covariance matrix for the fourth component is \( 3I_{8 	imes 8} \). Figure 3.8 shows the scatter plot for a
Table 3.4: Example 4. Discriminative measures $d_c(h)$, for $c = 1, \ldots, 4$. Calculations for $d_c(h)$ are based on the 8-dimensional Gaussian mixture with parameters estimated as posterior modes. The variable added at each step is listed with the corresponding values of $d_c(h)$ and RSM. The last variable to be included under the pre-specified stopping rules is underlined.

<table>
<thead>
<tr>
<th>Step</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_i$</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>3</td>
<td>6</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>$d_1(h)$</td>
<td>0.2200</td>
<td>0.0395</td>
<td>0.0208</td>
<td>0.0167</td>
<td>0.0175</td>
<td>0.0185</td>
<td>0.0194</td>
<td>0.0192</td>
</tr>
<tr>
<td>$x_i$</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>3</td>
<td>6</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>$d_2(h)$</td>
<td>0.4491</td>
<td>0.0754</td>
<td>0.0584</td>
<td>0.0566</td>
<td>0.0523</td>
<td>0.0525</td>
<td>0.0502</td>
<td>0.0417</td>
</tr>
<tr>
<td>$x_i$</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>$d_3(h)$</td>
<td>0.6216</td>
<td>0.0420</td>
<td>0.0424</td>
<td>0.0411</td>
<td>0.0414</td>
<td>0.0440</td>
<td>0.0487</td>
<td>0.0580</td>
</tr>
<tr>
<td>$x_i$</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>7</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>$d_4(h)$</td>
<td>0.3493</td>
<td>0.0247</td>
<td>0.0222</td>
<td>0.0211</td>
<td>0.0212</td>
<td>0.0229</td>
<td>0.0274</td>
<td>0.0352</td>
</tr>
<tr>
<td>$x_i$</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>5</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>RSM</td>
<td>0.2688</td>
<td>0.6009</td>
<td>0.6499</td>
<td>0.6529</td>
<td>0.6531</td>
<td>0.6532</td>
<td>0.6532</td>
<td>0.6533</td>
</tr>
</tbody>
</table>

randomly selected subset of the generated sample.

The Bayesian EM algorithm was used with up to 16 components using default, relatively vague priors. The model identified four Gaussian components; among them, three are of interest, as the rest is for background noise. According to the stopping rule, RSM selects four variables ($x_1, x_2, x_3, x_8$) only, and detected only three clusters in total. $d_c$ appropriately suggests that not all the dimensions play equally important roles in each cluster, e.g., $x_1$ and $x_8$ are not selected in the third cluster, however they are important in the first two clusters, $x_6$ is only included for the third cluster. Hoff (2006) developed a model-based clustering approach (MBSC) such that clusters differ from each other in terms of their means and/or variances in one or more dimensions. Therefore, even though the main goal in this approach is model-based clustering, it may also indicate relevant dimensions for each cluster. We implemented MBSC in this example using its default parameters as given in Hoff (2006), however, it only identified one cluster with all dimensions as relevant, so clearly being quite misleading in this example, which is really quite representative of the kinds of practical issues very often faced. In contrast, our discriminatory information analysis very
Figure 3.8: Pairwise scatter plots of a randomly selected subset of the $n = 6,000$ observations in Example 5.

adequately identifies cluster-specific discriminatory variable subsets.

3.4 Discriminatory marker variables in flow cytometry data analysis

Multi-parameter flow cytometry can measure up to 12-15 cell surface markers on thousands of cells per second and is a routine, and rapidly expanding, biological assay. The primary use of flow cytometry data is in identifying subpopulations in these large data sets that represent different regions of the multivariate marker space that relate to differentiation of cells and their biological function. Multivariate mixture models
Table 3.5: Example 5. Discriminative measures $d_c(h)$, for $c = 1, \ldots, 4$. Calculations for $d_c(h)$ are based on the 8-dimensional Gaussian mixture with parameters estimated as posterior modes. The variable added at each step is listed with the corresponding values of $d_c(h)$ and RSM. The last variable to be included under the pre-specified stopping rules is underlined.

<table>
<thead>
<tr>
<th>Step</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_i$</td>
<td>8</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>$d_1(h)$</td>
<td>0.1302</td>
<td>0.0695</td>
<td>0.0507</td>
<td>0.0497</td>
<td>0.0507</td>
<td>0.0559</td>
<td>0.0601</td>
<td>0.0617</td>
</tr>
<tr>
<td>$x_i$</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>$d_2(h)$</td>
<td>0.0245</td>
<td>0.0006</td>
<td>0.0001</td>
<td>0.00004</td>
<td>0.00003</td>
<td>0.00003</td>
<td>0.00002</td>
<td>0.00002</td>
</tr>
<tr>
<td>$x_i$</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>3</td>
<td>8</td>
<td>2</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>$d_3(h)$</td>
<td>0.0384</td>
<td>0.0065</td>
<td>0.0034</td>
<td>0.0020</td>
<td>0.0021</td>
<td>0.0024</td>
<td>0.0028</td>
<td>0.0036</td>
</tr>
<tr>
<td>$x_i$</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>8</td>
<td>7</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>RSM</td>
<td>0.3136</td>
<td>0.5881</td>
<td>0.6207</td>
<td>0.6412</td>
<td>0.6414</td>
<td>0.6394</td>
<td>0.6394</td>
<td>0.5804</td>
</tr>
</tbody>
</table>

are increasingly used (Chan et al., 2008) and the interest in identifying relevant subsets of markers, addressing the general questions posed above in Section 3.1, is fundamental to both analysis of experimental results and to the design and selecting of marker variables for future studies.

An example is given from a context where a subset of variables is known biologically to define a scientifically interesting subpopulation, as an applied proof-of-concept. The data come from a study of regulatory T cells, or Tregs, a specialized subtype of T cells that are critical to the maintenance of immune cell homeostasis and tolerance to self-antigens. The data set of more than 1 million observations contains multiple cellular subtypes of which Treqs are just one subpopulation. With $p = 13$ cell surface marker variables measured; the interest is in identifying Tregs within the heterogeneous cell population and the use of discriminative information in isolating this group. The 13 variables are labelled FSC-A, SSC-A, CD4, CD45RO, CD39, CD127, CD25, CD95, HLADR, FoxP3, CD152, CD3 and vAmineViolet. One 2–dimensional view of some particular non-Gaussian subtypes is shown in Figure 3.9.
Figure 3.9: Scatter plots of a small random subsample of the flow cytometry data on some important marker variable dimensions. The contours are those of the $2 \times 2$ dimensional margins on identified subpopulations $f_c(x)$ in the full $13 \times 13$ dimensional mixture; the area of each contour displayed is approximately proportional to corresponding estimated probabilities $\alpha_c$. This is based on the model with parameters estimated by the posterior mode, and non-Gaussian subtypes are defined using the mixture aggregation method as described in Section 2.4.

modes of attraction. Analysis using repeat posterior mode search identified a highest posterior mode at a model with $C = 54$ modes. Aggregating components by mode yields the fitted mixture that is used for discriminatory evidence evaluation. First, the subtype corresponding to Tregs was identified based on biological knowledge; Tregs are defined as having high values in FoxP3, CD25, CD39 and CD4 and low values in vAmineViolet and CD127. Let $c = T$ denote the Treg subpopulation and evaluate markers for their ability to discriminate Treg cells from the remaining 53 component subtypes in this fitted mixture. The value of $d_T(h)$ for all possible subsets $h \subseteq 1 : 13$ are computed; Table 3.6 reports a summary selection, including the most discriminatory subset of $k$ markers for each $k = 1, \ldots, 13$ along with two additional subsets of interest. Very good discrimination of Tregs can be obtained by using as
Table 3.6: Summary discriminative measures for selected marker variable subsets in the analysis of Tregs flow cytometry data, with variables: FSC-A, SSC-A, CD4, CD45RO, CD39, CD127, CD25, CD95, HLADR, FoxP3, CD152, CD3 vAmineViolet labeled from 1 to 13, respectively.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$d_T(h)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13</td>
<td>0.04</td>
</tr>
<tr>
<td>1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13</td>
<td>0.04</td>
</tr>
<tr>
<td>1, 2, 4, 5, 6, 7, 8, 10, 11, 12, 13</td>
<td>0.04</td>
</tr>
<tr>
<td>1, 2, 4, 5, 6, 7, 8, 11, 12, 13</td>
<td>0.04</td>
</tr>
<tr>
<td>1, 2, 4, 5, 6, 7, 8, 11, 12</td>
<td>0.04</td>
</tr>
<tr>
<td>1, 2, 4, 5, 6, 7, 8, 12</td>
<td>0.05</td>
</tr>
<tr>
<td>1, 2, 4, 5, 6, 7, 8</td>
<td>0.05</td>
</tr>
<tr>
<td>1, 2, 4, 5, 7, 8</td>
<td>0.06</td>
</tr>
<tr>
<td>2, 4, 5, 7, 8</td>
<td>0.08</td>
</tr>
<tr>
<td>2, 4, 5, 7</td>
<td>0.14</td>
</tr>
<tr>
<td>2, 5, 7, 8</td>
<td>0.15</td>
</tr>
<tr>
<td>2, 5, 8</td>
<td>0.36</td>
</tr>
<tr>
<td>5, 8</td>
<td>0.76</td>
</tr>
<tr>
<td>8</td>
<td>0.93</td>
</tr>
</tbody>
</table>

few as 6 marker variables, with only modest decrease of $d_T$ as additional markers are added; this is relevant for future studies that may aim to isolate only Treg cells and can rely on a reduced set of markers. In addition, the specific optimizing markers (1, 2, 4, 5, 7, 8) are FSC-A, SSC-A, CD45RO, CD39, CD25, CD95, and these are essentially those known biologically a priori to characterize Treg cells.

Posterior simulation generated a posterior sample of size 20,000, initialized at the posterior mode. From this posterior sample, the number of Gaussian mixture components has approximate posterior mean and median of 112, with a corresponding 95% HPD interval of (110, 114). The corresponding approximate posterior for the number of identifiable subtypes following modal aggregation yields mean 56.1, median 56 and 95% interval (53, 60). The posterior for $d_T(1 : 13)$ appears in Figure 3.10 to compare with the estimated value of 0.037 from Table 3.6.
Figure 3.10: Histogram of posterior samples for $d_T(1 : 13)$ in the analysis of flow cytometry data, with the modal estimate indicated by the vertical dashed line. Dotted line indicates posterior mean.

3.5 Discussion

The utility and effectiveness of the new discriminative information measure introduced here are demonstrated in both the synthetic and real data examples. With an inherent focus on discrimination between mixture components based on the overlap measure of concordance, the approach is intuitive and natural. Coupled with existing approaches and software tools for posterior mode search and posterior simulation in multivariate mixtures, the approach extends the toolbox of statistical discrimination and classification for studies aiming to dissect the roles played by variables, both individually and in association with other variables, in the determination of discrimination of mixture subpopulations. In contrast to variable selection approaches (e.g. Raftery and Dean, 2006; Kim et al., 2006), this new method overlays an existing mixture model analysis to explore and quantify the roles of subsets of variables, and
so can be applied easily and routinely across analysis. Further in comparison, the current approach is computationally accessible and scalable. The dissection of roles of variables cuts deeper than such methods in evaluating local discriminatory roles of variables; that is, assessing all possible subsets of variables for their roles on each subpopulation, rather than aiming to select one set of variables for all components; this is demonstrably key in applications such as the flow cytometry study illustrated here, where different, generally small subsets of variables can characterize subpopulations, with some variables being irrelevant of discrimination of many components but critically relevant for others.
Discriminative information measure in flow cytometry data analysis

The design of a panel of specific biomarkers to identify target cell subsets in flow cytometry can be difficult when specific markers unique to each cell subset do not exist, and a combination of parameters must be used to identify target cells of interest and exclude irrelevant events. In the absence of any standard for designing gating strategies, different laboratories may end up using very different panels and gating strategies to isolate the same target cell subset. In the context of highly standardized multi-center studies, local variations in parameter selection and gating strategies may result in increased variability and reduced reproducibility of the assay. Thus, the ability to objectively measure the contribution of a parameter or group of parameters towards target cell identification independent of any gating strategy could be very helpful for both panel design and gating strategy design. In this chapter, the discriminative information measure evaluation (DIME) developed in Chapter 3 provides an objective basis for inclusion or exclusion of specific parameters in a panel, and shows how ranked sets of such parameters can be used to optimize gating strategies. An
illustrative example of the application of DIME to streamline the gating strategy for a highly standardized carboxyfluorescein succinimidyl ester assay is described.

4.1 Introduction

Multi-parameter flow cytometry technology has seen dramatic advances in recent years, with 5 or more color assays now performed routinely in many basic and translational research laboratories. Standardization of all aspects of FCM, from instrument setup to data analysis, is an ongoing effort by multiple organizations, since standardization is necessary for consistent data comparison across sites.

In the drive to maximize recovery and purity (Loken et al., 1990), gating strategies can sometimes become increasingly complex even when there are relatively few parameters being measured. While this is effective for a single laboratory, it is difficult to apply complex gating strategies consistently across different instruments and operators across multiple laboratories. In the context of highly-standardized multi-center FCM studies, local variations in parameter selection and gating strategies may result in increased variability and reduced reproducibility of the assay (Maecker et al., 2005). Thus, the ability to objectively measure the contribution of a specific parameter or combination of parameters towards target cell identification independent of any gating strategy could be very helpful for both panel and gating strategy design.

Mixture model based analysis offers the ability to monitor target cell subsets directly in multiple dimensions without use of a sequence of gates. Several groups have recently published gating-free model-based approaches to cell subset identification using statistical mixtures of Gaussian, T or skewed distributions (Chan et al., 2008; Pyne et al., 2009; Boedigheimer and Ferbas, 2008; Lo et al., 2008). Here, I show that the predictive density resulting from such model-based approaches can be exploited to perform DIME for FCM parameters. The analysis evaluates parameter
usefulness for identifying a target cell subset that can be specified as some collection of mixture components. From a biological perspective, DIME provides insight into optimal parameter combinations that characterize a cell subset in a way that is independent of any particular gating strategy. Practically, it provides an objective basis for standardizing the analysis of flow cytometry panels in multi-center clinical trials, and can contribute to improved assay reproducibility. I show the application of DIME to the design of a simplified gating strategy for a carboxyfluorescein succinimidyl ester (CFSE)-based assay designed to measure CD4 and CD8 T lymphocyte proliferation following antigen challenge. The context for this proof-of-concept analysis was a three center pilot study (BD Biosciences, Universit de Montreal/NIML and Duke University) sponsored by DAIDS to standardize the assessment of T lymphocyte proliferation using a panel for CD3, CD4, CD8, CFSE and an amine viability stain. Experts at the three centers had, through careful evaluation of their collective data, developed a standard consensus gating strategy that was designed to reduce background and enhance detection of specific proliferation.

Under the CFSE data set, 8 parameters are measured: FSC-A, FSC-H, SSC-A, CFSE, CD4, Amine, CD3 and CD8. A gating strategy was designed taking into account the experience of the participating laboratories related to CFSE data analysis. The gates that were considered to be useful in order to reduce background, increase signal and allow for detection of aberrant samples were: doublet discrimination gate (FSC-H vs FSC-A), CFSE vs SSC-A, lymphocyte gate FSC-A vs SSC-A, CD3 vs SSC-A and CD4 vs CD8 (in order to discriminate all possible T cell subsets).

The original expert-determined standardized gating strategy made use of all 8 parameters. To explore alternative, unconventional gating sequences, I look for projections of the most informative subset of parameters as ranked by DIME in which the target clusters are well separated in the remaining parameters. Using this approach, a revised and considerably simplified manual gating scheme was found and
comparative evaluations of the original and revised strategies performed.

4.2 Evaluation and “gold standards”

The analysis also evaluates parameters based on sensitivity and specificity calculations relative to a chosen “gold standard” for cell subtype classification of all the data points. Using the full set of parameters, a cell with parameter values $x$ is classified as of subtype $c$ if $\alpha_c f_c(x|\theta_c)$ in equation (3.1) gives the maximum value over subtypes. When using all parameters, this is defined as the “gold standard” classification, i.e., a hypothetical “true” classification for the purposes of comparing classification based on fewer. Recomputing the classifications based on the marginal mixture on any reduced set of parameters, I can then evaluate which cells are misclassified with respect to each cell subtype relative to this gold standard. The resulting empirical sensitivity and specificity for any subset of parameters provides a useful practical guide to the impact of reducing to parameter subsets defined as optimal with respect to DIME.

4.3 Finding proliferating lymphocytes with Bayesian mixture modeling

The data are modeled using an 8-dimensional mixture of Gaussians with Bayesian EM for model fitting, followed by modal clustering to identify cell subsets described in Chapter 3. The Bayesian EM analysis identified 4 modal clusters corresponding to CD4 non-dividing, CD4 dividing, CD8 non-dividing and CD8 dividing T lymphocytes; Figure 4.1 shows the CD4/CFSE and FSC-A/SSC-A projections. Proliferating (CFSE low) cells are in orange, while non-proliferating (CFSE high) cells are in red. While the non-proliferating lymphocytes had classical scatter characteristics, the proliferating blasts were not in the standard “lymphocyte gate” location on FSC-A/SSC-A, and would be impossible to distinguish on the basis of scatter alone. Instead, they were identified on the basis of their CD3, CD4 and CD8 properties.
The vertical histogram on the left panel illustrates how the ability to separate the CD4 from CD8 lymphocytes is lost when the CD4 marker is excluded, but dividing and non-dividing cells can still be separated on CFSE alone. DIME captures quantitatively the loss of ability to separate any given cell subset from the others when a parameter or set of parameters is excluded.

4.4 Finding the optimal combination of $k$ parameters to identify target cell subsets

To identify parameter combinations that are best able to discriminate the target cell subsets (dividing and non-dividing CD4 and CD8 lymphocytes), DIME is calculated for all possible subsets of size $k$. Since there are 8 parameters and each can be absent or present, there are 255 ($2^8 - 1$) subsets to evaluate. The maximally informative
subsets of each size $k$, where $k$ ranged from 8 to 1, are shown in Figure 4.2. The

**Figure 4.2**: A schematic showing the parameter subsets with the best discriminative ability for each target cell subset.

parameter subset of size $k$ with the best discriminative ability for each of the target cell subsets is shown. The top panel shows the color and positional encoding for the glyph used to represent parameters in the analysis. The next 4 panels show the change in discrimination for each target cell subset when the best combination of $k$ markers is used (with $k$ going from 8 to 1 horizontally). For each value of $k$, there are $k$ parameters used to calculate DIME, chosen so that this particular combination gives the best discrimination out of all possible combinations of $k$ markers. The particular $k$ markers that give the best discrimination is indicated by the colored glyph, where included parameters are above the horizontal line, and excluded parameters below. For example, looking at the glyph in the Dividing CD4 panel (second from
(top), for $k = 7$, the parameter excluded is FSC-H. In other words, to identify dividing CD4 events with 7 parameters, our analysis suggests that the optimal parameters are (FSC-A, SSC-A, CFSE, CD4, Amine, CD3, CD8). Surprisingly, the exclusion of the T cell receptor complex component CD3, which is a classical marker for T lymphocytes, resulted in minimal loss of discriminative information for all 4 target subsets, indicating that the CD3 marker is statistically redundant in the presence of CD4 and CD8. FSC-H and the amine viability parameter also contribute little unique discriminative information for these subsets under the given experimental conditions.

A complementary use of DIME is to identify the critical discriminatory parameters by examining the sequence of parameters dropped for the minimally informative subsets of size $k$, as shown in Figure 4.3, which shows how rapidly the ability to discriminate events from these target cell subsets is degraded when critical parameters are dropped. The results show that CD4, CD8 are important for all subsets, and CFSE is critical for the dividing cell subsets, while the scatters are more important for non-dividing lymphocyte subsets.

4.5 Evaluating the drop in sensitivity and specificity when using fewer parameters

To validate DIME results, I evaluated the sensitivity, specificity and accuracy of classifying cells in each target cell subset by calculating the classification of every event in all 255 parameter combinations. The classification of events with all parameters present was used as the reference standard. In other words, I first assume that the classification of events into dividing CD4, non-dividing CD4, dividing CD8, non-dividing CD8 or “none of the above” subsets with all 8 parameters present is correct and hence serve as our reference cell subsets. I then calculate the number of true positives, false positives, true negatives and false negatives for each subset
Figure 4.3: A schematic showing the parameter subsets with the least discriminative ability for each target cell subset.

to derive the sensitivity, specificity and accuracy shown in Figure 4.4 for the most informative subsets identified. The figure shows the values for $k = 8$ to $k = 1$ optimal parameters using the same parameter sets shown in Figure 4.2, and confirms that sensitivity, specificity and accuracy are maintained at high levels and only begin to degrade rapidly below 4 parameters when the optimal parameter subsets given by DIME are used. Sensitivity is calculated as $\frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$, specificity as $\frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}}$ and accuracy as $\text{Subset Fraction} \times \text{Sensitivity} + \text{Specificity} \times (1 - \text{Subset Fraction})$. Figure 4.4 shows that this brute-force evaluation supports the usefulness of DIME since the optimal parameter subsets retain high values for these parameters down to subsets of size 4. An obvious advantage is that it is computationally trivial compared with
exhaustive evaluation by classification of events, and hence much more scalable to larger parameter subsets, as well as automatic.

4.6 Simplification of gating strategy

The information provided by DIME as to the relative usefulness of each parameter, and in particular, the revelation that CD3 did not contribute materially to the discrimination of proliferating and non-proliferating CD4+ and CD8+ lymphocytes, motivated us to explore alternative, unconventional gating strategies that excluded CD3, by simply looking for projections in which the target clusters were well separated in the remaining parameters. An extremely simple gating strategy is found that had only 3 gating generations using 5 parameters that was effective in separating the target cell clusters, and this was mapped to a manual gating strategy (Figure 4.5).

The reduced manual gating strategy was subsequently evaluated on all the data samples from a single CFSE standardization test panel performed by all 3 laboratories. The data samples were drawn from multiple patients treated under 3 differ-
Figure 4.5: Design of a simplified 3-generation gating strategy using 5 parameters suggested by use of DIME and cell subsets identified with mixture model analysis. In each step, the red labeled clusters were discarded and yellow labeled clusters retained.

dient stimulation conditions (unstimulated, CMV pp65 and anti-CD3/CD28 mAbs).

For all the quantities of interest (counts of CD4 and CD8 proliferating and non-
proliferating populations), the evaluated cell subset frequencies agreed with the orig-\ninal expert manual analysis performed at the NIML (Figure 4.6). While the results
suggest that consistent analysis can be achieved using different gating strategies, the
gating strategy suggested by DIME analysis is the simplest and hence a good can-
didate for multi-institution standardization studies when using this specific panel.

4.7 Discussion

I have described DIME for evaluation of the usefulness of any given variable or subset
of variables for discriminating a target cell subset. To the best of my knowledge, this
is the first description of an automated, quantitative approach to evaluating how
useful any given parameter in a flow cytometric panel is for identifying cell subsets
of interest. I demonstrate using a CFSE proliferation assay as an example that
such a measure is potentially useful for both panel optimization and gating strategy,
Figure 4.6: Comparison of simplified gating strategy (red symbols) with the original 9-generation gating strategy (blue symbols) on all the CFSE data sets used in a single experiment across 3 different labs (BD=square, Duke=triangle, NIML=circle) under 3 different treatment conditions (unstimulated, CMVpp65 peptide mix (15mers overlapping by 11aa) and anti-CD3/CD28 mAbs).

The principle underlying DIME is conceptually simple. In contrast to the sequential 2D approach of conventional gating, multi-dimensional statistical analysis identifies cell populations using all parameters simultaneously. In a fitted model, DIME basically formalizes the idea that if we can “drop” one or more variables at a time, and evaluate the change in discrimination of subsets, we will know how much the “dropped” variable contributes to target cell subset identification. DIME thus further extends the multi-dimensional analysis to provide a quantitative measure of the contribution each variable makes towards each identified cell subset of interest, giving an objective basis for panel and gating strategy design.

An objective, automatically computable measure of parameter usefulness has many benefits. In the context of clinical FCM assays in GCLP-compliant labora-
tories, DIME is useful for reducing the amount of trial-and-error in optimal gating strategy design by identifying the most informative parameters. As the number of parameters increases, the availability of DIME can significantly reduce the effort to find a target cell subset by identifying the subgroups of maximally informative parameters. The use of an objective basis for rationally designing a gating strategy is also likely to increase acceptance of that strategy by flow experts at different institutions. In turn, acceptance and use of a common gating strategy contributes to reducing the variability in multi-center studies. In some cases, DIME may reveal that certain parameters provide no additional information and can be dropped from the panel when cost is an issue. Panel reduction may also result in increasing the applicability of an assay by making the assay feasible on less sophisticated cytometers.

DIME can also be used to evaluate if different markers perform equivalently, and the potential loss or gain of sensitivity and specificity afforded with swapping markers. This can then be used to inform decisions on panel construction - for example, in cell sorting applications, it is necessary to use live cells, and hence a useful question that can be answered using discriminative measures could be “What is the impact of swapping an intracellular marker, that requires permeabilization and fixation to identify, with one or more cell surface markers, that may be used to identify viable cells?” Computationally, the discriminative measure provides a natural mechanism for feature selection, and I am currently investigating the extent to which this can be used for adaptive dimension reduction in clustering applications.

Fully automated gating techniques are currently being intensively researched at several institutions, but manual gating is likely to remain the standard practice for some time. The model-based approaches to automated cell subset identification can be complementary to manual gating, and provide useful information to guide both parameter selection and gating strategy design.
5

Hierarchical Bayesian mixture modeling

5.1 Introduction

In recent years, high dimensional large data sets have become common in various areas of application. For instance, in immune response studies and other areas of biology, statistical mixture modeling is becoming established for analysis of increasingly large data sets from flow cytometry technologies (e.g., Chan et al., 2008; Lo et al., 2008; Finak et al., 2009; Pyne et al., 2009; Manolopoulou et al., 2010). As discussed in Chapter 4, core interests lie in identifying and resolving multiple subtypes of immune cells, differentiated by the levels of activity (and presence/absence) of subsets of cell surface receptor molecules, as well as other biomarkers of cell phenotypes. FCM technology provides an ability to assay multiple single cell characteristics—reporters of cell surface receptors together with additional biomarkers—on many cells rapidly and cheaply; as a result, FCM is increasingly widely used and promoting increased interest in customization of statistical approaches to match evolving experimental biology techniques. The work in this chapter responds, in part, to one recent innovation—a combinatorial encoding method that leads to the ability to sub-
stantially increase the numbers of cell subtypes the method can, in principle, define. This new biotechnology, coupled with an interest in refined mixture modeling that respects the nature of FCM biomarker/reporter data and addresses key masking issues in using standard mixture approaches, provides the motivation for the statistical model development and application here.

Structured, hierarchical mixture models, with the applied goals of automated inference to identify specific cellular subtypes in very large samples of T-cells, are described in this chapter. The models represent natural, hierarchical partitioning of the multivariate sample space of flow cytometry data based on a partitioning of information from biomarker and cell surface receptor reporters. The hierarchical structuring leads, in the consequent Bayesian analysis, to conditional partitioning of large data sets based on biomarker profiles, and then within each partition a customized mixture model analysis of molecular reporter data. In the latter, model specification respects the biotechnological design by incorporating priors linked to the combinatorial encoding patterns. The partitioning provides recursive dimension reduction, resulting in efficient and precise mixture modeling analysis applied to smaller subsets of data across the hierarchy, while the combinatorial encoding based priors induce a focus on relevant parameter regions of interest.

The chapter is organized as follows. Section 5.2 discusses flow cytometry biomarker and molecular reporter data, and the new combinatorial encoding method. Section 5.3 introduces the novel mixture modeling strategy, discusses model specification and aspects of its Bayesian analysis. Section 5.4 includes detailed development of customized MCMC methods and use of GPU implementations of components of the analysis that can be parallelized to exploit desktop distributed computing environments for these increasingly large-scale problems. Section 5.5 provides an illustration using synthetic data simulated to reflect the combinatorial encoded structure. Section 5.6 discusses an application analysis in a combinatorially encoded validation
study of antigen specific T-cell subtyping in human blood samples, and Section 5.7 provides a summary of the chapter.

5.2 Flow cytometry in immune response studies

Much of immunology is concerned with understanding the cellular adaptive immune response mediated by T-cells, a subclass of lymphocytes. Many, functionally different subtypes of T-cells are characterized by differing cell surface markers (clusters of differentiation, CD markers) and the specificity of a given T-cell is determined by the clonally distributed T-cell receptor, various protein segments, or peptide epitopes, that are presented by larger Major Histocompatibility Complex (MHC) molecules. The vast range of unique TCRs means that any pathogen will have several peptide epitopes recognizable by some T-cell. Flow cytometry (FCM) uses fluorescent dyes tagged to molecular reporters to identify cell subsets. The typical use is to identify T-cells expressing a specific TCR by labeling the natural ligand (peptide-MHC) with a fluorescent dye and then detecting the cells that bind it via their cell surface receptors. In practice, multimers of peptide-MHC— involving with four or more peptide-MHC molecules— are used to increase binding strength and stability. Each color/dye defines a marker, or reporter, for the specific multimer; resulting FCM measurements are measured via laser excitation of the fluorescent intensities across, typically, millions of cells in a sample.

Importantly, FCM methods also generate additional biomarker measurements that reflect general cell characteristics, or phenotypes, related to biological function. Common biomarkers include several measures of light scattered from the surface of the cell, a multiplexed dump channel measurement that can be used exclude cells not of interest, and a measurement of cell viability that identifies dead cells. These biomarkers differentiate very large classes of T-cell subtypes themselves, though they provide limited information on specific, less prevalent subtypes that the multimer
binding defines within the broader categorization by biomarkers.

Current flow cytometers can discriminate around 12-15 different multimer reporters. Multimer labeling requires the use of one optical channel for each peptide epitope, and the optical spillover from one fluorescent dye into the detector channels for others—i.e., frequency interference—limits the number. This therefore severely limits the number of epitopes—corresponding to subtypes of specific T-cells—that can be detected in any one sample. In many applications, such as in screening for candidate epitopes against a pathogen or tumor to be used in an epitope-based vaccine, there is a need to evaluate many potential epitopes with limited sample availability. This represents a major current challenge to FCM as currently used.

Recently, techniques that exploit **combinatorial encoding of multimers** have been developed to address this challenge. Combinatorial encoding expands the number of antigen-specific T-cells that can be detected (Hadrup and Schumacher, 2010). The basic idea is simple: by using *multiple different fluorescent labels for any single epitope*, we can identify many more types of antigen-specific T-cells by decoding the color combinations of their bound multimer reporters. For example, using $k$ colors, we can in principle encode $2^k - 1$ different epitope specificities. In one strategy, all $2^k - 1$ combinations would be used to maximize the number of epitope specificities that can be detected (Newell et al., 2009). In a different strategy, only combinations with a threshold number of different multimers would be used to minimize the number of false positive events; for example, with $k = 5$ colors, we could restrict to only combinations that use at least 3 colors to be considered as valid encoding (Hadrup et al., 2009). As antigen-specific T-cells are typically exceedingly rare (often on the order of 1 in 10,000 cells), the robust identification of these cell subsets is a challenging statistical problem. Previous studies have established the feasibility of a 2-color encoding scheme; this paper describes statistical methods to automate the detection of antigen-specific T cells using data sets from a novel 3-color encoding
scheme.

Direct application of mixture models will typically generate imprecise if not unacceptable results due to the inherent masking of low probability subtypes. All standard statistical mixture fitting approaches suffer from masking problems that are increasingly severe in contexts of huge data sets in expanding dimensions. Estimation and classification results focus heavily on fitting to the bulk of the data, resulting in large numbers of mixture components being identified as modest refinements of the model representation of more prevalent subtypes (Manolopoulou et al., 2010). These approaches just does not have the ability to home-in on small features of the data reflecting low probability components or collections of components that together represent a rare biological subtype of interest. Hence, it is natural to seek hierarchically structured models that successively refine the focus into smaller, select regions of biological reporter space. The conditional specification of hierarchical mixture models now introduced does precisely this, and in a manner that respects the biological context and design of combinatorially encoded FCM.

5.3 Hierarchical mixture modeling

5.3.1 Data structure and mixture modeling issues

Begin by representing combinatorially encoded FCM data sets in a general form, with the following notation and definitions.

Consider a sample of size $n$ FCM measurements $x_i$, ($i = 1 : n$), where each $x_i$ is a $p-$vector $x_i = (x_{i1}, x_{i2}, \ldots , x_{ip})'$. The $x_{ij}$ are measured intensities of light of specific wavelengths; some are related to several functional FCM biomarkers, the rest to light emitted by the fluorescent reporters of multimers binding to specific receptors on the cell surface. As discussed above, both types of measure represent aspects of the cell phenotype that are relevant to discriminating T-cell subtypes. Denote the number of multimers by $p_t$ and the number of biomarkers by $p_b$, with $p_t + p_b = p$. 69
The elements of $x_i$ are also ordered so that $x_i' = (b_i', t_i')$ where $b_i$ is the lead subvector of biomarker measurements and $t_i$ is the subvector of fluorescent intensities of each of the multimers being reported via the combinatorial encoding strategy.

Figure 5.1: 3D scatter plots of a randomly selected subset of the FCM data of Section 5.6 on 3 reporters. Left: Qdot 655-A vs. Qdot 605-A vs. PE-A. Right: Qdot 655-A vs. APC-A vs. PE-A

Figure 5.1 shows a random sample of real data from a human blood sample validation study generating measures on $p_b = 6$ biomarkers and $p_t = 4$ multimers of key interest. The figure shows a randomly selected subset of the full sample projected into the 3D space of 3 of the multimer encoding colors. Note that the majority of the cells lie in the center of this reporter space; only a small subset is located in the upper corner of the plots. This region of apparent low probability relative to the bulk of the data defines a region where antigen-specific T-cell subsets of interest lie.

Traditional mixture models have difficulties in identifying low probability component structure in fitting large data sets requiring many mixture components; the inherent masking issue makes it difficult to discover and quantify inferences on the biologically interesting but small clusters that deviate from the bulk of the data. This can be shown in the $p = 10$ dimensional example using the most effective and
very widely used Bayesian approach based on Dirichlet process (DP) Gaussian mix-
tures (West et al., 1994; Escobar and West, 1995; Ishwaran and James, 2001; Chan
et al., 2008; Manolopoulos et al., 2010). To fit the DP model, a truncated mix-
ture with up to 160 Gaussian components is used, and the Bayesian EM algorithm
in subsection 2.3 is used to find the highest posterior mode from multiple random
starting points (Lin et al., 2011; Suchard et al., 2010). The estimated mixture model
with these plug-in parameters is shown in Figure 5.2. Many mixture components
are concentrated in the main central region, with only a few components fitting the
biologically important corner regions. To adequately estimate the low density corner
regions would require a huge increase in the number of Gaussian components and
an enormous computational search challenge, and is simply infeasible as a routine
analysis.

Figure 5.2: Data of Figure 5.1 on contours of 2-dimensional margins of fitted
10-dimensional DP Gaussian mixture.
5.3.2 Hierarchical model

I define a novel hierarchical mixture model specification that respects the biomarker/reporter structure of the FCM data and integrates prior information reflecting the combinatorial encoding underlying the multimer reporters. Using \( f(\cdot|\cdot) \) as generic notation for any density function, the population density is described via the compositional specification

\[
f(x_{1:n}|\Theta) = \prod_{i=1}^{n} f(b_i|\Theta)f(t_i|b_i, \Theta)
\]

where \( \Theta \) represents all relevant and needed parameters.

This naturally focuses on a hierarchical partition: (i) consider the distribution defined in the subspace of biomarkers first, to define understanding of substructure in the data reflecting differences in cell phenotype at that first level; then (ii) given cells localized– and differentiated at this first level– based on their biomarkers, understand subtypes within that now based on multimer binding that defines finer substructure among T-cell features.

5.3.3 Mixture model for biomarkers

Heterogeneity in biomarker space is represented via a standard truncated Dirichlet process mixture model (Ishwaran and James, 2001; Chan et al., 2008; Manolopoulou et al., 2010; Suchard et al., 2010). A mixture model at this first level allows for first-stage subtyping of cells according to biological phenotypes define by the biomarkers alone. That is,

\[
f(b_i|\Theta) = \sum_{j=1}^{J} \pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j})
\]

where \( \pi_{1:J} \) are the component probabilities, summing to 1, and \( N(b_i|\mu_{b,j}, \Sigma_{b,j}) \) is the density of the \( p_{b} \)-dimensional Gaussian distribution for \( b_i \) with mean vector \( \mu_{b,j} \)
and covariance matrix $\Sigma_{b,j}$. The parameters \{${\pi}_{1..J}, \mu_{b,1..J}, \Sigma_{b,1..J}$\} are elements of the overall parameter set $\Theta$. Priors on these parameters are taken as standard; that for $\pi_{1..J}$ is defined by the usual \textit{stick breaking representation} inherent in the DP model, and I adopt proper, conditionally conjugate normal-inverse Wishart priors for the \{${\mu}_{b,j}, \Sigma_{b,j}$\}. First, $\pi_1 = \nu_1$, $\pi_j = (1 - \nu_1) \cdots (1 - \nu_{j-1}) \nu_j$, where $\nu_{j, j \neq J} \sim Be(1, \alpha_b)$, $\nu_J = 1$, and the hyper-parameter $\alpha_b \sim Ga(e_b, f_b)$ for some specified $e_b$ and $f_b$. Second, independently of the $\pi_j$ the normal mean and variance matrices are independent across components with priors
\[(\mu_{b,j}, \Sigma_{b,j}) \sim N(\mu_{b,j} | m, \lambda \Sigma_{b,j}) IW(\Sigma_{b,j} | \delta_b, \Phi_b)\]
for some specified hyper-parameters $m, \lambda, \delta_b, \Phi_b$.

The mixture model can be interpreted as arising from a clustering procedure depending on underlying latent indicators $z_{b,i}$ for each observation $b_i$. That is, $z_{b,i} = j$ indicates that biomarker vector $b_i$ was generated from mixture component $j$, or $b_i | z_{b,i} = j \sim N(b_i | \mu_{b,j}, \Sigma_{b,j})$, and with $P(z_{b,i} = j) = \pi_j$. The mixture model also has the flexibility to represent non-Gaussian T-cell region densities by aggregating a subset of Gaussian densities. This latter point is key in understanding that Gaussian mixtures do not imply Gaussian forms for biological subtypes, and is used in routine FCM applications with traditional mixtures (Chan et al., 2008; Finak et al., 2009).

Bayesian analysis using Markov chain Monte Carlo (MCMC) methods augments the parameter space with the set of latent component indicators $z_{b,i}$ and generates posterior samples of all model parameters together with these indicators. Over the course of the MCMC the $z_{b,i}$ vary to reflect posterior uncertainties, while conditional on any set of their values the data set is conditionally clustered into $J$ groups (some of which may, of course, be empty) reflecting a current set of distinct subpopulations; some of these may reflect one unique biological subtype, though realistically they generally reflect aggregates of subtypes that may then be further evaluated based on the multimer reporters. This is the key point that underlies the second component
of the hierarchical mixture model, as follows.

5.3.4 Conditional mixture models for multimers

Reflecting the biological reality, I posit a mixture model for multimer reporters \( t_i \), again utilizing a mixture of Gaussians for flexibility in representing essentially arbitrary non-Gaussian structure; we again note that clustering several Gaussian components together may overlay the analysis in identifying biologically functional subtypes of cells. I assume a mixture of at most \( K \) Gaussians, \( N(t_i|\mu_{t,k}, \Sigma_{t,k}) \), for \( k = 1 : K \). The locations and shapes of these Gaussians reflects the localizations and local patterns of T-cell distributions in multiple regions of multimer. However, recognizing that the above development of a mixture for biomarkers has the inherent ability to subdivide T-cells into up to \( J \) subsets, there is a need to reflect that the relative abundance of cells differentiated by multimer reporters will vary across these biomarker subsets. That is, the weights on the \( K \) normals for \( t_i \) will depend on the classification indicator \( z_{b,i} \) were they to be known. Since these indicators are part of the augmented model for the \( b_i \), I therefore condition on them to develop the model for \( t_i \).

Specifically, I take the set of \( J \) mixtures, each with \( K \) components, given by

\[
f(t_i|z_{b,i} = j, b_i, \Theta) = \sum_{k=1}^{K} \omega_{j,k} N(t_i|\mu_{t,k}, \Sigma_{t,k})
\]

where the \( \omega_{j,k} \) sum to 1 over \( k = 1 : K \) for each \( j \). As discussed above, the component Gaussians are common across biomarker subsets \( j \), but the mixture weights \( \omega_{j,k} \) vary and may be very different.

This leads to the natural theoretical development of the conditional density of multimer reporters given the biomarkers, defining the second components of each
term in the likelihood function of eqn. (5.1). This is
\[
f(t_i|b_i, \Theta) = \sum_{j=1}^{J} f(t_i, z_{b,i} = j|b_i, \Theta)
\]
\[
= \sum_{j=1}^{J} P(z_{b,i} = j|b_i, \Theta) f(t_i|z_{b,i} = j, b_i, \Theta)
\]
\[
= \sum_{j=1}^{J} \left\{ \frac{\pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j})}{f(b_i|\Theta)} \right\} \sum_{k=1}^{K} \omega_{j,k} N(t_i|\mu_{t,k}, \Sigma_{t,k}) \tag{5.3}
\]
\[
= \sum_{k=1}^{K} \omega_{i,k}(b_i) N(t_i|\mu_{t,k}, \Sigma_{t,k}) \tag{5.4}
\]
where
\[
\omega_{i,k}(b_i) = f(b_i|\Theta)^{-1} \sum_{j=1}^{J} \omega_{j,k} \pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j}). \tag{5.5}
\]
Notice that the \(\omega_{i,k}(b_i)\) are mixing weights for the \(K\) multimer components as reflected by eqn. (5.4); the model induces latent indicators \(z_{t,i}\) in the distribution over multimer reporter outcomes conditional on biomarker outcomes, with \(P(z_{t,i} = j|b_i) = \omega_{i,k}(b_i)\). These multimer classification probabilities are now explicitly linked to the biomarker measurements and the affinity of the datum \(b_i\) for component \(j\) in biomarker space.

From the viewpoint of the main applied focus on identifying cells according to subtypes defined by both biomarkers and multimers, key interest lies in posterior inferences on the subtype classification probabilities
\[
P(z_{b,i} = c, z_{t,i} = c|x_i, \Theta) \propto \sum_{(j,k) \in I_c} \pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j}) \omega_{i,k}(b_i) N(t_i|\mu_{t,k}, \Sigma_{t,k}), \tag{5.6}
\]
for each subtype \(c = 1 : C\). Here
\[
P(z_{b,i} = j, z_{t,i} = k|x_i, \Theta) \propto \pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j}) \omega_{i,k}(b_i) N(t_i|\mu_{t,k}, \Sigma_{t,k}), \tag{5.7}
\]
for \(j = 1 : J, k = 1 : K\), and where the index sets \(I_c\) contains biomarker and multimer component indices \(j\) and \(k\), respectively. These classification subsets and
probabilities will be repeatedly evaluated on each observation \( i = 1 : n \) at each iterate of the MCMC analysis, so building up the posterior profile of subtype classification.

One next aspect of model completion is specification of priors over the \( J \) sets of probabilities \( \omega_{j,1:K} \) and the component means and variance matrices \( \{ \mu_{t,1:K}, \Sigma_{t,1:K} \} \). This is done using the structure of a standard hierarchical extension of the truncated DP model (Teh et al., 2006). Under a prior from this class, the \( \omega_{1:t,1:K} \) are naturally independent of the \( \{ \mu_{t,1:K}, \Sigma_{t,1:K} \} \), and are also naturally linked across biomarker components \( j \); the specification of \( p(\omega_{1:j,1:K}) \) are detailed below.

First, generate a \( K \)-vector of probabilities \( \eta_{1:K} \) via the stick-breaking construction

\[
\eta_k = \phi_k \prod_{l=1}^{k-1} (1 - \phi_l), \quad k = 1, \ldots, K - 1,
\]

where \( \phi_k \sim \text{Beta}(1, \gamma_t) \), \( k = 1, \ldots, K - 1 \) and \( \phi_K = 1 \), and where \( \gamma_t \sim G(e_t, f_t) \) for some given hyper-parameters \( e_t, f_t \).

Then, for each biomarker component \( j = 1 : J \), generate the multimer mixture weights \( \omega_{j,1:K} \) via

\[
\omega_{j,k} = \phi_{j,k} \prod_{l=1}^{k-1} (1 - \phi_{l,j}), \quad k = 1, \ldots, K - 1,
\]

where \( \phi_{j,k} \sim \text{Beta}(\alpha_t \eta_k, \alpha_t (1 - \sum_{l=1}^{k} \eta_l)) \), \( k = 1, \ldots, K - 1 \), and \( \phi_{j,K} = 1 \). I use hyper-priors \( \alpha_t \sim G(a, c) \) for given hyper-parameters \( a, c \). The \( \Sigma_{t,1:K} \) are taken as independent of the other parameters and with \( \Sigma_{t,k} \sim IW(\Sigma_{t,k} \mid \delta_t, \Phi_t) \) for some specified \( \delta_t, \Phi_t \), corresponding to the usual conditionally conjugate prior.

The remaining aspect of the prior specification is that for \( \mu_{t,1:K} \), the multimer model component location vectors, and it is here that the structure of the combinatorial encoding design comes into play.
5.3.5 Priors on multimer component location vectors

The levels of different multimers represented by subtype means $\mu_{t,1:K}$ must be structured to reflect the combinatorial design. For any given epitope, reported fluorescent intensity levels are recognized as distributed around zero for cells lacking the corresponding cell surface receptor, in a range of low non-zero values, or at rather higher levels for cells targeted by the reporter. I capture this through a prior on the $\mu_{t,1:K}$ linked to corresponding regions in reporter space, structured to also capture the prior knowledge implicit in the strategy of multimer combinatorial encoding.

Define anchor regions in the $p_t$-dimensional multimer reporter space by a set of $R = 3^p_t$ anchor points, as follows. Represent by $0/L/H$ anchor points in any one multimer dimension, choosing specific values of $L, H$ on the reporter scale. Set $R = 3^{p_t}$ and define the set of $R$ vectors $m_{1:R}$ via

$$m_r = (m_{1,r}, m_{2,r}, \ldots, m_{p_{t},r})', \quad r = 1 : R,$$

where $m_{i,r} \in \{0, L, H\}$ and the $m_r$ vectors represent all distinct $R = 3^{p_t}$ combinations of 0, $L, H$ for each of the $p_t$ reporters. Effectively, the $m_r$ identify all $R$ subregions of the $p_t$-dimensional reporter space according to possible combinations of absent, low levels and high levels of each of the multimers being reported. For example, in the simplest case with $p_t = 2$, then $R = 8$, $m_r$ vectors are the columns of the matrix

$$
\begin{pmatrix}
0 & 0 & 0 & L & L & L & H & H \\
0 & L & H & 0 & L & H & 0 & L & H
\end{pmatrix}.
\]

Given the anchor vectors $m_{1:R}$, the prior for $\{\mu_{t,1:K}, \Sigma_{t,1:K}\}$ is now defined based on the following idea. We expect to see cell subtypes in a selection of the $R$ regions linked to anchor points, and as earlier anticipate that distributions of reporters within subtypes may be heterogeneous. Hence any one subtype may be represented by a number of the $\mu_{t,k}$ that are clustered within one of the $R$ regions, so that the resulting aggregate of the corresponding subset of the weighted $N(t_i|\mu_{t,k}, \Sigma_{t,k})$ distributions
reflects the reporter distribution for that cell subpopulation. This means a relevant prior for the $\mu_{t,k}$ will engender such clustering in the anchored regions reporter space while allowing for variability more globally. The natural model for this is to take the $\mu_{t,k}$ to be independent with marginal priors

$$
\mu_{t,k} \sim \sum_{r=1}^{R} q_r N(\mu_{t,k}|m_r, Q_r)
$$

for some variance matrices $Q_r$ where, as a default, we take $q_r = 1/R$, for $r = 1 : R$. In addition to allowing for the above described scientific clustering, this also allows for some or many of the $R$ anchored regions to be “empty” in the sense that none of the $\mu_{t,k}$ are generated from the corresponding $N(\cdot|m_r, Q_r)$ component of this mixture prior.

Specification of the $3 \times 3$ variance matrices $Q_r$ defines the expected levels of variation, and patterns of covariation, within a subset of the $\mu_{t,k}$ allocated to anchor region $r$. The default specification we make, following a broad study of the impact of variation in the values chosen is to base this on an overall scalar variance $q$ and a set of specified pairwise correlations that relate to the anchor regions. For the latter, high abundance of two specific multimers—represented by $H, H$—is consistent with positive correlation in the corresponding elements of $Q_r$; low abundance of one and high abundance of the other—i.e., $L, H$—is consistent with negative correlation; lack of correlation is relevant when either one of the multimers is absent, i.e., $0, X$ for any $X \in \{0, L, H\}$. As an example when $p_t = 3$, for the 3 anchor regions $r = s, u, v$ defined by $m_s = (H, L, H)'$, $m_u = (0, L, L)'$ and $m_v = (0, 0, H)'$, we take

$$
Q_s = q \begin{pmatrix} 1 & \rho_p & \rho_n \\ \rho_n & 1 & \rho_n \\ \rho_p & \rho_n & 1 \end{pmatrix},
Q_u = q \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & \rho_p \\ 0 & \rho_p & 1 \end{pmatrix},
Q_v = q \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},
$$

respectively, where $q$ controls overall levels of variation and $\rho_p, \rho_n$ are specified positive and negative correlations. Following studies to evaluate specification, we take
\( \rho_p = 0.6 \) and \( \rho_n = -0.6 \) as a default. The remaining \( Q_r \) matrices are filled out similarly corresponding to their anchor regions.

The specific anchor values of \( L, H \) are chosen to reflect known ranges of mean levels of low/high fluorescent intensities. This could be generalized to allow differing values that are specific to epitopes, and it is also possible to extend the Bayesian analysis to allow for uncertainty in these values by treating them as hyper-parameters. Standardized multimer measurements range from \(-4\) to \(10\). Though the specific ranges differ somewhat across multimer, we take \( L = -4 \) and \( H = 6 \) for all multimers, defining prior ranges that allow for all experienced data regions. Similar comments apply to choice of values for the \( Q_r \), in that the above specification might be relaxed by treating the \( \rho_p, \rho_n \) as hyper-parameters or even endowing each \( Q_r \) with, say, an inverse Wishart hyper-prior. Such extensions may be explored further in future in new applications. However, our current studies suggest that these extensions are overkill and unlikely to materially impact the resulting inferences; the specifications above have been customized to the known characteristics of FCM fluorescent reporter scales and I have evaluated a range of prior specifications and find strong levels of robustness to these specifications. The reasons for this are that the model already allows for uncertainty via the prior variability of the \( \mu_{t,1:K} \) around the means \( m_r \), and overlays this with an ability to add multiple \( \mu_{t,k} \) to any anchor region to fill-out a conditional mixture defining a flexible representation of the reporter distribution for the cell subtype in that region. That is, the model already has substantial degrees-of-freedom in adapting to observed data configurations.

5.4 Posterior computations

5.4.1 Augmented model and MCMC

Posterior computations use customized MCMC methods involving a combination of Gibbs sampling and Metropolis-Hastings. The overall strategy is standard in
Bayesian computation, involving augmentation of the model parameter space by sets
of mixture component indicators that (i) enable simulation of relevant conditional
distributions for model parameters, and (ii) are themselves then imputed from rele-
vant conditional posteriors as the MCMC proceeds. Thus the posterior simulations
for model parameters and mixture component indicators are jointly obtained, the
latter feeding into follow-on inferences on subtype classification for each cell, among
other things.

An outline to the augmentation ideas and the overall MCMC strategy is noted
here, with full technical details given in Sections 5.4.2, 5.4.3 and 5.4.4. The full
parameter set \( \Theta \) is

\[
\Theta = \{ (\mu_{b,1:j}, \Sigma_{b,1:j}, \pi_{1:j-1}, \alpha_b), (\mu_{t,1:K}, \Sigma_{t,1:K}, \omega_{1:j,1:K}, \alpha_t, \gamma_t) \}
\]

where the first subset relates to the biomarker mixture model and the second to that
for the multimers. In the first, subset, \( \alpha_b \) is a hyper-parameter underlying the DP
prior for the biomarker model whose role and prior are as defined in Section 5.3.3;
similarly, the hyper-parameters \( \alpha_t, \gamma_t \) of the multimer hierarchical DP model have
roles and priors defined in Section 5.3.4.

The augmented model includes the biomarker mixture component indicators \( z_{b,1:n} \)
earlier introduced as well as additional indicators underlying the hierarchical DP
mixture for multimer mixture components conditional on the \( z_{b,1:n} \).

5.4.2 Update component indicator variables

For each \( i = 1 : n \) in parallel due to conditional independence, update \( z_{b,i} \) and \( z_{t,i} \)
by sampling from their conditional multinomial (number of trials = 1 in each case)
posteriors defined by probabilities as follows:

\[
P(z_{b,i} = j | \ldots ) \propto \pi_j N(b_i | \mu_{b,j}, \Sigma_{b,j}), \quad j = 1 : J;
\]

\[
P(z_{t,i} = k | \ldots ) \propto \omega_{i,k}(b_i) N(t_i | \mu_{t,k}, \Sigma_{t,k}), \quad k = 1 : K.
\]
As $z_{b,i}$ is conditionally independent of $z_{t,i}$, sampling the multimer model indicators is done in parallel with the biomarker indicators.

5.4.3 Update biomarker model parameters

1. Update biomarker mixture weights and hyperparameter. Sampling mixture probabilities $\pi_{1:J}$ and the hyperparameter of the DP model use a Metropolis-Hastings extension of the standard component distributions (Ishwaran and James, 2001; Ji et al., 2009), as follows. The mixture probabilities $\pi_j$ are obtained from underlying beta variates $\nu_{1:J-1}$ as detailed in Section 5.3.3. Hence new $\pi_{1:J}$ samples are computed directly from resampled $\nu_{1:J-1}$ samples. For the latter, we have

$$p(\nu_j \ldots) \propto Be(\nu_j | a_{b,j} + 1, \alpha_b + \sum_{s=j+1}^J a_{b,s}) \prod_{k=1:K, i : z_{t,i} = k} \omega_{i,k}(b_i).$$

where $a_{b,j} = \sum_{i=1}^n 1_{z_{b,i} = j}$ for each $j = 1 : J - 1$. The complications here are that, since the $\pi_j$ are functions of the $\nu_j$, then $\nu_j$ is implicitly involved in both numerator and denominator terms of the product expression multiplying the base beta distributions. Hence a Metropolis-Hastings sampler is used for this step, based on a customized proposal distribution

$$\nu_j^* \sim Be(\nu_j | a_{b,j} + a_{u,j} + 1, \alpha_b + \sum_{s=j+1}^J a_{b,s} + a_{u,s})$$

with

$$a_{u,s} = \sum_{i=1}^n 1_{u_{t,i} = s}, \quad j = 1 : J - 1$$

$$(u_{t,i} = s) = \max_{r=1:J} \pi_r N(b_i | \mu_{b,r}, \Sigma_r) \omega_{r,z_{b,i}}$$

This proposal distribution is an approximation of $p(\nu_j \ldots)$ by taking off the denominator $f(b_i | \Theta)^{-1}$ in the product expression and assuming that $\pi_{u_{t,i}} N(b_i | \mu_{u_{t,i}}, \Sigma_{u_{t,i}}) \omega_{u_{t,i}, z_{b,i}}$ dominates the rest of the component values. My experience with examples and the
data analysis reported is that this generates acceptable convergence with acceptance rates for these components of the MCMC around 10-50%.

The weights \( \pi_1:J \) are then evaluated by the formula of its stick-breaking representation in Section 5.3.3. Next, the hyperparameter \( \alpha_b \) is resampled from

\[
(\alpha_b|\ldots) \sim Ga(J + e_b - 1, f_b - \sum_{j=1}^{J-1} \log(1 - \nu_j)).
\]

2. Update biomarker component means and variance matrices. For each \( j = 1:J \) the mean \( \mu_{b,j} \) has conditional posterior

\[
p(\mu_{b,j}|\ldots) \propto N(\mu_{b,j}|\bar{\mu}_b, \bar{\Sigma}_b) \prod_{k=1:K, i:z_{t,i} = k} \omega_{i,k}(b_i)
\]

with

\[
\bar{\mu}_{b,j} = c_j (m/\lambda + \sum_{i:z_{b,i} = j} b_i) \quad \text{and} \quad \bar{\Sigma}_b = c_j \Sigma_{b,j}
\]

where \( c_j = \lambda/(1 + \lambda a_{1,j}) \). Again a Metropolis-Hastings sampler is needed for this step as the base normal distribution here is multiplied by a term that depends in complicated ways on \( \mu_{b,j} \). I use the customized proposal distribution \( \mu_{b,j}^* \sim N(\mu_{b,j}^*|\bar{\mu}_b, \bar{\Sigma}_b) \)

where

\[
\bar{\mu}_{b,j}^* = c_j^* (m/\lambda + \sum_{i:z_{b,i} = j} b_i + \sum_{i:u_{t,i} = j} b_i) \quad \text{and} \quad \bar{\Sigma}_b^* = c_j^* \Sigma_{b,j}
\]

with \( c_j^* = \lambda/(1 + \lambda a_{1,j} + \lambda a_{u,j}) \).

A similar structure and MCMC strategy arises for each of the \( j = 1:J \) variance matrices \( \Sigma_{b,j} \); the conditional posteriors are

\[
p(\Sigma_{b,j}|\ldots) \propto IW(\Sigma_{b,j}|\delta_b + a_{b,j} + 1, \Phi_b + \Psi_{b,j}) \prod_{k=1:K, i:z_{t,i} = k} \omega_{i,k}(b_i)
\]

where

\[
\Psi_{b,j} = (\mu_{b,j} - m)(\mu_{b,j} - m)'/\lambda + \sum_{i:z_{b,i} = j} (b_i - \mu_{b,j})(b_i - \mu_{b,j})'.
\]
I use the customized proposal distribution

$$\Sigma_{b,j}^* \sim IW(\Sigma_{b,j}^*|\delta_b + a_{b,j} + a_{u,j} + 1, \Phi_b + \Psi_{b,j}^*)$$

with

$$\Psi_{b,j}^* = \Psi_{b,j} + \sum_{i:u_i=j} (b_i - \mu_{b,j})(b_i - \mu_{b,j})'$$.

The pair \((\mu_{b,j}, \Sigma_{b,j})\) are updated together each iterate. This generates acceptable convergence with acceptance rates for these components of the MCMC around 20-45%.

5.4.4 Update multimer model parameters

1. Update multimer mixture weights and hyperparameter. With the definitions and notation of the multimer mixture model parameters of Section 5.3.4, the logic and details of the MCMC steps are as follows.

For each \(k = 1:K\) \(\phi_k\) has conditional posterior

$$p(\phi_k|\ldots) \propto Be(\phi_k|1, \gamma_t) \prod_{j=1;J,k=1;K} Be(\phi_{j,k}|\alpha_t \eta_k, \alpha_t (1 - \sum_{l=1}^k \eta_l))$$

To choose a proposal distribution, first, for each \(i = 1:n\) and independently over \(i\), generate a set of auxiliary indicator variables \(q_i\) from conditional multinomials on \(k = 1:K\) cells with number of trials= 1 and

$$(q_i = k) \propto \eta_k N(t_i; \mu_{t,k}, \Sigma_{t,k}), \quad k = 1:K.$$  

Given these sampled values, generate

$$(\phi_k^*|\ldots) \sim Be(a_{q,k} + 1, \gamma_t + \sum_{h=k+1}^K a_{q,h}), \quad k = 1:K - 1,$$

where \(a_{q,r} = \sum_{i=1}^n 1_{q_i=r}\) for each \(r = 1:K\). It generates acceptable convergence with acceptance rates for these components of the MCMC around 10-40%.

The sets of weights \(\omega_{j,k}\) and the \(\eta_{1,K}\) probabilities are then evaluated by the
formulæ given in Section 5.3.4. Further, the hyper-parameter $\gamma_t$ is resampled from
\[(\gamma_t|\ldots) \sim Ga(K + \epsilon_t - 1, f_t + \sum_{k=1}^{K-1} \log(1 - \phi_k)).\]

Next, for each $j = 1 : J$ and $k = 1 : K$, the latent probabilities $\phi_{j,k}$ of subsection 5.3.4 have conditional posterior
\[p(\phi_{j,k}|\ldots) \propto Be(\phi_{j,k}|\alpha_t \eta_k, \alpha_t(1 - \sum_{l=1}^{k} \eta_l)) \prod_{k=1; i : z_{t,i} = k} \omega_{i,k}(b_i)\]

I use the customized proposal distribution
\[(\phi_{j,k}|\ldots) \sim Be(\alpha_t \eta_k + g_{j,k}, \alpha_t(1 - \sum_{h=k+1}^{K} \eta_h) + \sum_{h=k+1}^{K} g_{j,h});\]
where $g_{j,k} = \sum_{i=1}^{n} 1_{z_{t,i} = k, a_t_i = j}$. It generates acceptable convergence with acceptance rates for these components of the MCMC around 5 – 50%.

2. Update multimer component means and variance matrices. For each $k = 1 : K$ the mean $\mu_{t,k}$ is sampled using an additional auxiliary random quantity that allocates the multimer to one of the $K$ anchor regions based on current parameters and indicators. That is, for each $k$ independently, draw an auxiliary indicator $\tau_k$ from the multinomial with one trial and probabilities on $k = 1 : K$ given by
\[P(\tau_k = r|\mu_{t,k}^{-}, \ldots) \propto q_r N(\mu_{t,k}^{-}|m_r, Q_r), \quad k = 1 : K.\]
Then draw $(\mu_{t,k}|C_k = r, \ldots) \sim N(\mu_{t,k}|m_{t,k}, M_{t,k})$ where
\[m_{t,k} = M_{t,k}(Q_r^{-1} m_r + \Sigma_{t,k}^{-1} \sum_{i : z_{t,i} = k} t_i) \quad \text{and} \quad M_{t,k}^{-1} = Q_r^{-1} + a_{t,k} \Sigma_{t,k}^{-1}\]
with $a_{t,k} = \sum_{i=1}^{n} 1_{z_{t,i} = k}$.

Finally, resample the variance matrices from
\[(\Sigma_{t,k}|\ldots) \sim IW(\Sigma_{t,k}|\delta_t + a_{t,k}, \Phi_t + \sum_{i : z_{t,i} = k} (t_i - \mu_{t,k})(t_i - \mu_{t,k})').\]
5.4.5 Post-MCMC analysis

MCMC fitting of mixture models suffer from the well-known label switching problem, complicating posterior inference. I address this using the state-of-the-art method for relabeling MCMC samples described and implemented in Cron and West (2011).

At iterate $s$ of the MCMC analysis with a current set of all model parameters $\Theta^{(s)}$ and sets of mixture component indicators generically denoted by $Z^{(s)}$, this method relabels components in each of the mixtures: first for $f(b_i|\Theta)$ and then for $f(t_i|b_i, \theta)$.

The computationally efficient and statistically effective relabeling strategy aims to match labels between MCMC iterates, so links the labels at iterate $s$ with those at $s - 1$, to best match the assignments of all $n$ observations to labeled mixture components between the two steps. The developed structured extension of mixture models require a stagewise application of the strategy. Components of the biomarker model $\sum_{j=1}^{J} \pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j})$ are relabeled first based on the biomarker indicator matching; the relevant subset of the parameters are relabeled accordingly. Then, relabeling is applied to the multimer model $\sum_{k=1}^{K} \omega_{i,k}(b_i) N(t_i|\mu_{t,k}, \Sigma_{t,k})$ with the consequent re-ordering of the relevant parameters. Each of these is a straight application of the method of Cron and West (2011), and posterior inferences follow based on the sets of relabeled parameters.

Given the relabeled set of parameters for the hierarchical mixture model of equation (5.1), I follow previous work (Chan et al., 2008; Finak et al., 2009) and my development in earlier chapters here, in defining subtypes by aggregating proximate components $\pi_j N(b_i|\mu_{b,j}, \Sigma_{b,j})\omega_{i,k}(b_i) N(t_i|\mu_{t,k}, \Sigma_{t,k})$. That is, if a number of components cluster together and contribute to defining a mode in the mixture in one region of marker space, they are identified as a group and their renormalized average is taken as defining a subtype. This allows for a clear definition of subtypes, that may have quite non-Gaussian shapes, and is implemented by first identifying modes.
in the mixture of equation (5.1), and then associating each individual component with one mode based on proximity to the mode. An encompassing set of modes is first identified via numerical search; from some starting value \( x^0 \), iterative mode search is performed using the BFGS quasi-Newton method for updating the approximation of the Hessian matrix, and the finite difference method in approximating gradient, to identify local modes. This is run in parallel from \( JK \) initial values \( x^0 = (\mu_{b,j}^t, \mu_{t,k}^t)^t, j = 1 : J, k = 1 : K \), and results in some number \( C \leq JK \) unique modes. Grouping components into clusters defining subtypes is then done by associating each of the mixture components with the closest mode, i.e., identifying the components in the basin of attraction of each mode.

5.4.6 Computational implementation

The MCMC implementation is naturally computationally demanding, especially for larger data sets as in the FCM applications. Profiling the developed MCMC algorithm indicates that there are three main aspects that take up more than 99% of the overall computation time when dealing with moderate to large data sets as in FCM studies. These are: (i) Gaussian density evaluation for each observation against each mixture component as part of the computation needed to define conditional probabilities to resample component indicators; (ii) the actual resampling of all component indicators from the resulting sets of conditional multinomial distributions; and (iii) the matrix multiplications that are needed in each of the multivariate normal density evaluations. However, as previously shown in standard DP mixture models (Suchard et al., 2010), each of these problems is ideally suited to massively parallel processing on the CUDA/GPU architecture (graphics card processing units). In standard DP mixtures with hundreds of thousands to millions of observations and hundreds of mixture components, and with problems in dimensions comparable to those here, that reference demonstrated CUDA/GPU implementations providing speed-up of several
hundred-fold as compared with single CPU implementations, and are dramatically superior to multicore CPU analysis.

With this in mind, the implementation for the new hierarchical mixture models here also exploit massive parallelization and GPU implementation. The code takes advantage of the ease of use of the Matlab programming/user interface, via Matlab scripts dealing with the non-computationally intensive parts of the MCMC analysis, while a Matlab/Mex/GPU library serves as a compute engine to handle the dominant computations noted above in a massively parallel manner. Some other aspects of the implementation of the library code include storing persistent data structures in GPU global memory to reduce the overheads that would otherwise consume a significant amount of time to transfer data between the Matlab CPU memory space and the GPU global memory space. In examples with dimensions comparable to those of the studies here, this library and the customized code delivers speed-up comparable to the pure CUDA C/GPU implementation as indicated in Suchard et al. (2010).

5.5 Simulation study

I now use the above developed hierarchical mixture model for an illustrative example. A sample of size 10000 with \( p = 8 \) dimensions was drawn such that the first 5 dimensions was generated from a mixture of 7 normal distributions. The last two normal distributions have approximate equal mean vectors \( (0, 5.5, 5.5, 0, 0)' \), \( (0, 0, 0, 0)' \), and common diagonal covariance matrix \( 2I \) with component proportions 0.02 and 0.01. The remaining normal components have very different mean vectors and larger variances compared with the last two normal components. So \( b_i \) is the subvector of the first 5 dimensions, with \( p_b = 5 \). The last three dimensions are generated from a mixture of 10 normal distributions, where only two of them have high mean values across all three dimensions. The component proportions vary according to which normal component \( b_i \) was generated from. So \( t_i \) is the subvector of the last three
dimensions, and $p_t = 3$. The data was designed to have a distinct mode such that all the five dimensions $b_2, b_3, t_1, t_2$ and $t_3$ are of positive values, the rest are negative. The cluster of interest with size 140 is indicated in red in Figure 5.3.

![Figure 5.3: Pairwise scatter plots and one 3-dimensional scatter plot of simulated data. The cluster of cells of interest is plotted in red. Here $b_i$ is the $i$th dimension in the biomarker subvector, and $t_i$ is the $i$th dimension in the tetramer subvector. The lower right plot indicates that the cluster of interest is the outer layer of the sample under $b_2, t_1$ and $t_2$. The other 3 subplots shows that the cluster is hard to identify with traditional 2-D gating.](image)

I first fit the sample with the standard DP Gaussian mixture model. Analysis allows up to 64 components using default, relatively vague priors, so encouraging smaller components. The Bayesian EM algorithm was run repeatedly from many random starting points; the highest posterior mode identified 14 Gaussian components. Using parameters set at this mode leads to an estimated posterior classification prob-
ability matrix for the entire sample. The cluster representing the synthetic subtype of interest was completely masked as is shown in Figure 5.4.

Figure 5.4: Scatter plots of synthetic data example as in Figure 5.3. Using standard DP mixture analysis, a relatively large subtype is identified. Cells assigned to this subtype are colored blue, and this blue region extends to include much of the actual subtype region, in red, in this synthetic data set. The analysis is unable to identify the correct subtype region.

I then contrast the above with results from analysis using the new hierarchical mixture model. Model specification uses $J = 10$ and $K = 16$ components in biomarker and multimer model components, respectively. In the biomarker model, priors favor smaller components: I take $e_b = 50, f_b = 1, m = 0, \delta_b = 26, \Phi_b = 10I$.

Similarly, under multimer model, I choose $e_t = 50, f_t = 1, \delta_t = 24, \Phi_t = 10I, L = -4, H = 6$. I constructed $m_{1:R}$ and $Q_{1:R}$ for $\mu_{t,k}$ following Section 5.3.5, with $q = 5$, $\rho_p = 0.6$ and $\rho_n = -0.6$. The MCMC computations were initialized based on the
specified prior distributions. Across multiple numerical experiments, I have found it useful to initialize the MCMC by using the Metropolis-Hastings proposal distributions as if they are exact conditional posteriors—i.e., by using the MCMC as described but, for a few hundred initial iterations, simply accepting all proposals. This has been found to be very beneficial in moving into the region of the posterior, and then running the full accept/reject MCMC thereafter. This analysis saved 20,000 MCMC draws for summary inferences. Global visuals addressing MCMC convergence, such as the trace plots of some log-likelihood components exemplified in Figure 5.5, are encouraging.

![Figure 5.5](image)

**Figure 5.5**: Trace plot over the last 5,000 MCMC iterates of computed values of the model of equation (5.1) on seven randomly selected data points.

After relabeling and aggregating components based on the parameters at the last iterate this identified $C = 29$ modes in this “current” posterior sample. The posterior classification probabilities of eqn. (5.6) were computed for the last 3,000 iterates, and data classification based on the resulting approximate posterior means. As shown in
Figure 5.6, the hierarchical model analysis correctly identified 133 observations out of the 140 target sample.

Figure 5.6: Scatter plots of synthetic data example as in Figures 5.3 and 5.4. The cluster of interest is plotted in red, and almost wholly overlays that identified by the hierarchical mixture model in blue.

5.6 Study of data from human blood samples

Peripheral blood samples were obtained from healthy volunteers for validation studies of the combinatorial encoding strategy. Peripheral blood mononuclear cells (PBMC) were labeled using the encoding strategy described; that is, with a mixture of fluorescent reporters indicating cell phenotype in biomarker space and ability to recognize specific peptide-MHC epitopes in multimer space. The data set comprises \( n = 752,835 \) samples cells in \( p = 10 \) dimensions; the 10 measured features are the
$p_b = 6$ biomarkers labeled FSC-A, FSC-H, SSC-A, Dump FITC-A, CD8 and Viability APC-Cy7-A, and the $p_t = 4$ multimers labeled Qdot 655-A, Qdot 605-A, APC-A and PE-A. The primary interest is to detect T-cells specific for cytomegalovirus (CMV), Ebstein-Barr virus (EBV) and influenza (Flu) virus peptides with the following combinatorial encoding scheme, where high intensities of the multimers in each define the T-cell subtype in reporter space:

CMV = (PE-A, Qdot 655-A, Qdot 605-A),
EBV = (PE-A, APC-A, Qdot 655-A),
Flu = (PE-A, Qdot 605-A, APC-A).

**Figure 5.7:** Reference plot indicating clusters/subtypes of interest in the T-cell human blood data. A subset of the data is shown in grey, CD8+ T cells are plotted in yellow, CMV, EBV and FLU groups are plotted in green, red, and blue, respectively. These latter three identified subtypes contained 2739, 66 and 59 observations respectively. Subtypes of interest are CD8+ and positive also in the three corresponding multimer features.

A subset of the data on some of the key features was already noted in Figure 5.1 in discussion of small probability structure of biologically interesting cell subtypes. Figure 5.7 illustrates the events determined to be positive for the targeted tetramer
combinations for CMV, EBV and FLU using a standard manual gating procedure that is used as a reference plot for comparing with the model-based analysis here.

Model specification uses \( J = 100 \) and \( K = 100 \) components in the biomarker and multimer model components, respectively. These are expected to be encompassing values with the model intrinsically able to cut-back to lower, data-relevant values based on the Bayesian DP mixture structure. In the biomarker model component, priors favor larger numbers of smaller components: \( e_b = 50, f_b = 1, m = 0_{p_b \times 1}, \lambda = 5, \delta_b = p_b + 1 + 10, \Phi_b = 10I \). Similarly, for the multimer model, \( e_t = 50, f_t = 1, \delta_t = p_t + 1 + 20, \Phi_t = 10I, L = -4, \) and \( H = 6 \). We constructed \( m_{1:R} \) and \( Q_{1:R} \) for \( \mu_{t,k} \) following Section 5.3.5 with \( q = 5, \rho_p = 0.6 \) and \( \rho_n = -0.6 \). The MCMC computations were initialized as detailed in the study of the synthetic data above and run for a total of 15,000 iterates. Posterior classification probabilities based the last 1,000 iterate are used, again with exploration of visual diagnostics of convergence, e.g., Figure 5.8.

The MCMC analysis outputs deliver the opportunity to enquire about a broad range of model characteristics; these include aspects of the mixture structure over biomarkers, while the primary biological focus rests on characteristics of the mixture structure over multimers and the classification of cells according to subtypes in multimer space. Some aspects of the former are worth noting initially. The fitted model indicates that there are approximately 1021 modes in the distribution. Contour plots of the estimated model in selected dimensions in Figure 5.9 show that a smaller number of Gaussian components can now represent the sample space much more effectively than with the original model as depicted in Figure 5.2.
Figure 5.8: Trace plot over the last 1,000 MCMC iterates of model density values on five randomly selected data points in analysis of human blood data.

Figure 5.9: Scatter plots of the real data. The contours are the 2-dimensional margins on each Gaussian component in the full 10-dimensional mixture.
The MCMC analysis also delivers posterior samples of the $z_{b,i}$ and $z_{t,i}$ themselves; these are useful for exploring posterior inferences on the number of effective components out of the maximum (encompassing) value $JK$ specified. Clusters that have high intensities for multimer combinations mapping to the multimer encodings are identified and shown in Figure 5.10. The estimated CMV, EBV and FLU groups contains 12, 3 and 11 product of Gaussian components, respectively. The structured, hierarchical mixture model can flexibly capture many smaller Gaussian components as well as over-coming the masking issues of standard approaches. Some of the modes here have as few as 10 observations out of original 752,835 observations, reflecting the ability of the hierarchical approach to successfully identify quite rare events of potential interest.
Figure 5.10: Identified clusters of interest. A subset of the data is shown in grey. CMV, EBV and FLU groups are plotted in green, red, and blue respectively. The identified three groups contain 2849, 4 and 216 observations respectively. The group with 4 events does not meet the cut-off of 10 events and 0.002% of CD8 for positive events we have suggested (Andersen et al., 2012) and may represent false positive or background events.

5.7 Discussion

I have defined and explored a novel class of structured, hierarchical mixture models with the applied goals of automated inference to identify specific cellular subtypes in very large samples of T-cells. The approach (i) involves a natural, model-based hierarchical partitioning of FCM biomarker and multimer reporter measurements, and (ii) integrates a second stage hierarchical prior for the latter customized to the new biotechnological design of combinatorial encoding of multimers. The first step (i) represents key aspects of the biological reality: important cell subtypes
defined by cell surface receptor function – as reported by the multimer data – are differentially represented across what is typically a large number of subtypes defined by biomarkers. Model-based stratification in biomarker space effectively leads to sample dimension reduction that can overcome the inherent challenges of estimating what are typically low subtype probabilities. The second step (ii) addresses the specific features introduced in the recently proposed encoding method, a method that can greatly increase the number of T-cell antigen specificities distinguishable in limited biological samples using flow cytometry.

Combinatorial encoding is set to impact broadly on FCM studies by allowing a huge increase in the numbers of cell types detectable. This is particularly relevant in screening of optimal peptide epitopes in many areas of immunology, including vaccine design in cancer and other areas, where the diversity of potential antigen-specific T cell subsets is substantial. Using conventional FCM methods with one fluorescent marker for each multimer-complex would require the collection and analysis of large (and infeasible) volumes of peripheral blood from each patient, and the sample sparing advantages of combinatorial encoding are key to a feasible screening strategy. Previous studies have shown the practicality of a dual encoding scheme (Hadrup et al., 2009; Hadrup and Schumacher, 2010; Andersen et al., 2012; Newell et al., 2009); this study significantly extends the earlier work to demonstrate the feasibility of a triple encoding scheme. As the number of antigen-specificities detectable increases exponentially with the degree of the encoding scheme, moving from a 2–color to 3– or more color encoding scheme has a disproportionate impact.

However, because standard visual gating is performed on 2D projections, the manual analysis of higher-dimensional encoding schemes is extremely challenging and highly labor-intensive due to the need to take patient heterogeneity into account. Hence there is an obvious need for automated detection of combinatorially encoded multimer binding events. Traditional mixture models simply lack the ability
to identify the very small and subtle subtype structure of combinatorially encoded multimer events when applied to very large data sets due to masking by large background components. This is a key feature of and benefits arising from the new model: as demonstrated in the example, it is by design able to identify and quantify subpopulation structure related to relatively rare cell subtypes, i.e., to generate fitted models in which low probability mixture components are appropriately located in weakly populated regions of the $p$–dimensional sample space, and that are essentially undetectable using standard mixture approaches.

Part of the cost in application of the new, customized class of models is the implied computational burden; the structured MCMC is quite expensive in that respect. Efficient computational implementations are key, and we have developed coding strategies to maximally exploit the inherent opportunities for within MCMC parallelization customized to GPU processors. The code is optimized for CUDA/GPU processing with an accessible Matlab front-end (provided under an open source license) for implementing the model analysis as presented.
6

Bayesian tree-based mixture models

6.1 Introduction

Compared with traditional DPM models, the hierarchical mixture models of Chapter 5 demonstrate an ability to identify low probability component structures in fitting large data sets, thereby overcoming the inherent masking issue. This ability is partially achieved by partitioning the data dimensions into two parts according to the underlying problem context; hence, partitioning underlies dimension reduction, resulting in an efficient and precise mixture modeling analysis. In this chapter, I continue to explore the general question of learning mixture models by dividing the data dimensions into low dimensional subspaces.

Data in high dimensions is often difficult to understand and visualize. The emergence of various new application domains, such as bioinformatics and text modeling, are characterized by high dimensions. Two common challenges are the curse of dimensionality and model overfitting. The specificity of dissimilarities between points in a high dimensional space diminishes, rendering many clustering models ineffective because the model becomes vulnerable to the presence of noise. As dimensions
increase, models frequently require an exponential increase in the number of parameters, a requirement that easily leads to overfitting. Several approaches have been developed to engender parameter constraints and, thus, control increases in parameter dimensions. Graphical models are frequently used to address these problems, taking advantage of the (conditional) independencies between subsets of variables based on their representations using a graph. Dobra et al. (2004) focused on modeling a sparse covariance structure in a high dimensional graphical model, such that one dimension can be predicted well by a relatively small subset of the other dimensions. Among graphical models the class of mixture trees, which is a subset of graphical models proposed by Meila and Jordan (2000), admits tractable inference and learning algorithms and has demonstrated its ability for practical applications; here the variable set is partitioned into subsets (clusters) on which tree structure estimations can be performed independently. Attias (2001) proposed a hierarchical, modular mixture model with separate mixtures for different groups of variables and additional higher level mixtures for the residual dependencies between groups. However, the learning method described by Attias (2001) considers only a fixed model structure and cannot be used to infer variable grouping.

Unlike previous graphical model approaches, I propose here a new Bayesian mixture model using binary trees, constructed with the goal of modeling the data structure from each individual dimension. The dependencies among the dimensions are captured by the tree structure. Kingman’s coalescent (Section 1.5.1) is utilized as a prior for the tree structure. An efficient MCMC algorithm is developed for posterior inference on model parameters and tree structures. The chapter is organized as follows. Section 6.2 introduces the Bayesian mixture of binary trees with Kingman’s coalescent prior. Section 6.3 includes a detailed development of the customized MCMC methods for posterior sampling. Section 6.4 discusses selected simulation studies to illustrate the model, and Section 6.5 provides a summary.
6.2 Bayesian mixture of binary trees

This section introduces a mixture of binary trees model. Consider a sample of size \( n \) denoted by \( x_i, (i = 1 : n) \), where each \( x_i \) is a \( p \)-vector \( x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})' \). Instead of placing a binary tree structure on the observations (Teh et al., 2008) to perform agglomerative hierarchical clustering based on the pair-wise distances among observations, a binary tree is placed on the dimensions to model the interactions among variables. Because a tree structure can represent only pairwise dependencies (differences), it has limited ability to infer location, i.e., the tree structure will remain the same for data having the same covariance matrix regardless of the mean shift (the same amount of shift in every dimension). This limitation of the tree structure has no impact on hierarchical clustering schemes; however, by directly placing a tree structure on the data dimensions, the resulting allocation clustering scheme will only distinguish observations based on their covariance structures.

This limitation can easily be resolved by representing \( x_i \) by \( x_i = \mu + \epsilon_i \), where \( \mu \) represents the mean location of the data, and \( \epsilon_i \) is the residual for each observation. Thus, a binary tree with \( p \) leaves is used to model the covariance structure of \( \epsilon_{ij}, j = 1, \cdots, p \). Define \( t = \{t_1, \cdots, t_p-1\} \) and \( \pi = \{\pi_1, \cdots, \pi_{p-1}\} \) as the vector of branching times and the set of partitions defined in Section 1.5.1, respectively. The value of \( t \) reflects the dependencies between two merged sets; a small value of \( t \) represents a larger correlation between two merged sets. \( \pi \) groups variables into dependent subsets. Let Coalescent\((p)\) represent the distribution (equation 1.4) over binary tree structures according to Kingman’s coalescent process. A Bayesian model for the residuals can be written as

\[
\epsilon_i | t, \pi \sim p(\epsilon_i | t, \pi),
\]

\[
t, \pi \sim \text{Coalescent}(p),
\]

(6.1)

for some specified density \( p(\cdot | \cdot) \).
Both Teh et al. (2008) and Henao and Lucas (2012) have demonstrated that using the message passing algorithm (Bishop, 2006) to marginalize recursively from the leaves to the root of the tree, the density function for $\epsilon_i$ in equation (6.1) can be written in a recursive form. The details of deriving the marginal density of $\epsilon_i$ are provided by Henao and Lucas (2012). According to Figure 6.1, define $z_k \in \pi_i$ be a node in a binary tree, and $z_{c_1}, z_{c_2}$ be its children. Then define $p(z_k|z_c, t)$ to be the transition density between a child node, $z_c$, and its parent, $z_k$. Then $q(z_k|t, \pi)$, the density of $z_k$, can be recursively defined as

$$q(\epsilon_{ij}|t, \pi) = \delta_{\epsilon_{ij}}, \quad j = 1, \cdots, p,$$

$$q(z_k|t, \pi) = \prod_{z\in C} \int p(z_k|z_c, t)q(z_c|t, \pi)dz_c$$

$$= Z_k(\epsilon_i, t, \pi)q'(z_k|t, \pi)$$

where $C = \{c_1, c_2\}$ contains the two sets in $\pi_{k-1}$ that merge in $\pi_k$, $q'(z_k|t, \pi)$ is a density (integrating to 1) and $Z_k$ is the appropriate scaling factor. Assume a Gaussian distribution for the transition probability with the mean centered at the child node and the variance represented by the time elapsed between to two nodes,
where $\Delta_c$ is the time elapsed between $t_k$ and $t_c$. Denote the time at which the set $c$ was created as $t_c$. Then
\[
p(z_k|z_c, t_{1:k}) = N(z_k|z_c, \Delta_c)
\]
where $\Delta_c$ is the time elapsed between $t_k$ and $t_c$. Denote the time at which the set $c$ was created as $t_c$. Then
\[
q(z_k|\pi_k, t_{1:k}) = N(z_k|m_{c_1}, \tilde{s}_{c_1})N(z_k|m_{c_2}, \tilde{s}_{c_2})
\]
\[
= Z_k(\epsilon_i|\pi_k, t_{1:k})N(z_k|s_k(\tilde{s}_{c_1}^{-1}m_{c_1} + \tilde{s}_{c_2}^{-1}m_{c_2}), s_k)
\]
where $m_{c_1}$ and $s_{c_1}$ are the mean and the variance of $q(z_{c_1}|\cdot)$, respectively, with $\tilde{s}_{c_1} = \Delta_c + s_{c_1}$, and where $\Delta_{c_1} = t_k - t_{c_1}$. This leads to $s_k^{-1} = \tilde{s}_{c_1}^{-1} + \tilde{s}_{c_2}^{-1}$ and the normalization constant is
\[
Z_k(\epsilon_i|\pi_k, t_{1:k}) = (2\pi)^{-1/2}v_k^{-0.5} \exp\left(-\frac{(m_{c_1} - m_{c_2})^2}{2v_k}\right)
\]
where $v_k = \tilde{s}_{c_1} + \tilde{s}_{c_2} = 2\Delta_k + r_k$ and $r_k = 2t_{k-1} - t_{c_1} - t_{c_2} + s_{c_1} + s_{c_2}$. At the root,
\[
q(z_{p-1}|\pi_{p-1}, t_{1:p-1}) = N(z_{p-1}|m_{c_1}, \tilde{s}_{c_1})N(z_{p-1}|m_{c_2}, \tilde{s}_{c_2})N(z_{p-1}|\mu_z, s_z)
\]
\[
= Z_{p-1}(\epsilon_i|\pi_{p-1}, t_{1:p-1})N(z_{p-1}|s_{p-1}(\tilde{s}_{c_1}^{-1}m_{c_1} + \tilde{s}_{c_2}^{-1}m_{c_2}), s_{p-1})N(z_{p-1}|\mu_z, s_z)
\]
\[
= Z_{p-1}(\epsilon_i|\pi_{p-1}, t_{1:p-1})Z(\epsilon_i|\pi_{p-1}, t_{1:p-1})
\]
\[
N(z_{p-1}|(\tilde{s}_{c_1}^{-1}m_{c_1} + \tilde{s}_{c_2}^{-1}m_{c_2})/(s_{p-1}^{-1} + s_z^{-1}), (s_{p-1}^{-1} + s_z^{-1})^{-1}),
\]
where $N(z_{p-1}|\mu_z, s_z)$ is the initial distribution for the root of the tree. Hence, the density function for $\epsilon_i$ in equation (6.1) can be written as
\[
p(\epsilon_i|t, \pi) = Z(\epsilon_i|\pi_{p-1}, t_{1:p-1}) \prod_{k=1}^{p-1} Z_k(\epsilon_i|t, \pi),
\]
where $Z_k$ is the normalization constant developed above, and $Z$ is the last normalization constant with
\[
Z(\epsilon_i|\pi_{p-1}, t_{1:p-1}) = (2\pi)^{-1/2}(s_{p-1}^{-1} + s_z^{-1})^{-0.5}
\]
\[
\exp\left(-\frac{(s_{p-1}(\tilde{s}_{c_1}^{-1}m_{c_1} + \tilde{s}_{c_2}^{-1}m_{c_2}) - \mu_z)^2}{2(s_{p-1} + s_z)}\right).
\]
Combining equation (6.2) and the prior in equation (1.4), it gives the joint prob-
ability for posterior inference:
\[ p(\epsilon, \pi, t) = Z(\epsilon|\pi, t) \prod_{k=1}^{p-1} \text{Exp}(-(p - k + 1)(p - k)\Delta_k/2)Z_k(\epsilon|\pi, t). \] (6.3)

### 6.2.1 Initial distribution specification

The initial distribution for the root, denoted by \( p(z_{p-1}) \), plays an important role in ensuring model identifiability. Under a flat initial distribution e.g. \( p(z_{p-1}) \propto 1 \), equation (6.2) reduces to
\[ p(\epsilon_1|t, \pi) = \prod_{k=1}^{p-1} Z_k(\epsilon_1|t, \pi). \] (6.4)

Because the kernel of each \( Z_k \) is an exponential function of the difference of means between two child nodes at merging point \( k \), the likelihood will remain the same for both \( \mu \) and \( \mu + d1_{p \times 1} \), given that the other parameters are fixed. Here, \( d \) can be any real number, i.e., equation (6.4) has no unique global mode with respect to \( \mu \).

This likelihood nonidentifiability is well illustrated in Figure 6.2, where a 2-dimensional data set of size 100 is generated according to a normal distribution with mean 0, variance 1 and 0.6 correlation. Because the true mean is 0, the tree structure is estimated directly based on the data, using the algorithm developed in Section 6.3.3. The log of the likelihood values (6.2) are computed for both \( \mu_1 \) and \( \mu_2 \) to vary from \(-1\) to \(1\) with 0.1 increments, with \( p(z_{p-1}) \) selected as normal with mean 0 and variance 100. Figure 6.2 illustrates that the log-likelihood is flat, in the sense that there are many possible solutions for \( \mu \) that correspond to the mode of likelihood. Hence, with no prior information for \( \mu \), the posterior distribution of \( \mu \) will be flat as well. The noninformative initial distribution has limited control over where the tree can be grown.

The above analysis indicates that \( z_{p-1} \) requires an informative initial distribution to avoid the model nonidentifiability issue. In an ideal case in which data originates
Figure 6.2: 3-dimensional and 2-dimensional plots of log-likelihood (6.2) values under flat initial distribution, with $z_{p-1} \sim N(0, 100)$.

from a single distribution with very small variance, the residual sum of squares for each dimension should be close to zero, so should be their parent node. Hence, $z_{p-1}$ should be restricted to be very close to zero, with the degrees of belief controlled by the variance of the initial distribution. Placing such an informative initial distribution on $z_{p-1}$ is analogous to likelihood regularization. Figure 6.3 reveals that an informative $p(z_{p-1})$ resolves the issue; the mode of the likelihood corresponds to the mean values near zero.

6.2.2 Mixture of binary trees model

The above development is used to model data with a single binary tree. Under heterogeneous populations, different sets of tree structures and mean vectors are required to model different subpopulations. Define latent indicators $u_i$ for each observation such that conditioning on $u_i = s$, $x_i = \mu_s + \epsilon_i$, and the structure of $\epsilon_i$ is captured by tree $s$. This process defines a mixture of binary trees. Let $W_s$ be the prior probability that $p(u_i = s)$, defined to follow the truncated DP stick-breaking construction. Let $S$ denote the maximum number of mixture components; the prior
under this mixture of binary trees is then

\[
\begin{align*}
\alpha & \sim Ga(e, f), \\
\nu_s & \sim Be(1, \alpha), \quad s = 1, \ldots, S - 1, \quad \nu_S = 1, \\
W_s & = (1 - \nu_1) \cdots (1 - \nu_{s-1})\nu_s, \\
\mu_s & \sim N(m, \tau I_{p \times p}), \\
\pi, t & \sim \prod_{k=1}^{p-1} \exp(- (p - k + 1)(p - k)\Delta_k/2).
\end{align*}
\]

This mixture of binary trees model is a specific class of DPM models with the ability to identify fewer than the maximum specified number of components, providing an automatic indicator of the effective number of components. Unlike the DPM models, the component-specific distributions are induced by the tree structure on the variables. The covariance structures among the variables are implicitly represented by the trees, while the covariance structure is explicitly represented by the covariance matrix using the DPM with Gaussian component distributions. This difference rep-
resents an advantage of the tree-based model which uses fewer parameters than the usual DPM models. More specifically, with \( p \)-dimensional data, each component in the mixture of trees model requires \( 3p - 3 \) parameters, a linear function with respect to dimension. However, each Gaussian component requires \( (p^2 + 3p)/2 \) parameters, a quadratic function with respect to dimension. For high dimensional data, the number of parameters in a DP Gaussian mixture model may become prohibitively large, whereas the number of parameters in the mixture of binary trees model increases only linearly with dimension.

The covariance structure may be inferred from the tree structure. Under the tree constructed above, the difference between two nodes connected by an edge of length \( \Delta \) is normally distributed with zero mean and

\[
\text{var}(z_k - z_c) = \Delta_c = \text{var}(z_k) + \text{var}(z_c) - 2\text{cov}(z_k, z_c).
\]

Hence, if \( \Delta_c \) is small relative to \( \text{var}(z_k) + \text{var}(z_c) \), the two nodes are positively correlated. If \( \Delta_c \) equals \( \text{var}(z_k) + \text{var}(z_c) \), the two nodes are uncorrelated. If \( \Delta_c \) exceeds \( \text{var}(z_k) + \text{var}(z_c) \), the two nodes are negatively correlated. In other words, positively correlated nodes will be merged first, followed by uncorrelated nodes; negatively correlated nodes are merged last.

6.3 Posterior computations

Posterior computations use customized MCMC methods involving a combination of Gibbs sampling, sequential Monte Carlo (SMC) sampling and Metropolis-Hastings. The overall strategy involves augmentation of the model parameter space by the set of mixture component indicators \( u_i \) that enable the simulation of relevant conditional distributions for model parameters, and are themselves imputed from relevant conditional posteriors as the MCMC proceeds. Thus, the posterior simulations for the
model parameters and mixture component indicators are jointly obtained.

The full parameter set Θ is Θ = {α, W_{1:S−1}, μ_{1:S}, π_{1:S}, t_{1:S}}. Elements α, W_{1:S−1} are updated using Gibbs sampling, μ_{1:S} are sampled according to a customized Metropolis-Hastings, and {π_{1:S}, t_{1:S}} are updated using an SMC sampling scheme as developed by Henao and Lucas (2012).

6.3.1 Update component indicator variables

For each i = 1 : n in parallel, update z_i by sampling from their conditional multinomial (number of trials = 1 in each case) posteriors defined by probabilities

\[ p(u_i = s | ...) \propto W_s Z_s(x_i - μ_s | π_{p−1}, t_{1:p−1}) \prod_{k=1}^{p−1} Z_{s,k}(x_i - μ_s | π_s, t_s), \quad s = 1 : S. \]

6.3.2 Update mixture probabilities and hyperparameter of the DP prior

Sampling mixture probabilities W_{1:S} and the hyperparameter of the DP prior uses standard Gibbs sampling. Specifically,

\[ ν_s | ... \sim Be(ν_s | 1 + a_s, α + \sum_{r=s+1}^{S} a_r), \]

\[ α | ... \sim G(S + e - 1, f - \sum_{r=1}^{S-1} \log(1 - ν_s)), \]

where \( a_s = \sum_{i=1}^{n} 1_{u_i = s} \) for each \( s = 1 : S \).

6.3.3 Update tree structures

Sampling the tree structures uses the SMC algorithm developed by Henao and Lucas (2012) based on equation (6.3). For cluster s, for a set of M particles, the posterior of \{t, π\} is approximated using a weighted sum of point masses obtained iteratively by drawing coalescing times t_k and chain states π_k one at a time from their posterior
as
\[
p(\Delta_{s,k}, \pi_{s,k} | t_{s,1:k}, \pi_{s,k-1}) = Z_{s,k,C}^{-1} \exp(-(p - k + 1)(p - k)\Delta_k/2)
\]
\[
= Z_{s,k}(x_{i,u_i=s} - \mu_s | \pi_{s,k}, t_{s,1:k}),
\]  
(6.5)
where \(Z_{s,k,C}^{-1}\) is the normalizing constant and \(C\) is a pair of elements of \(\pi_{k-1}\).

Equation (6.5) can be expanded as:
\[
p(\Delta_{s,k}, \pi_{s,k} | t_{s,1:k}, \pi_{s,k-1})
\]
\[
\propto Z_{s,k}(x_{i,u_i=s} - \mu_s | \pi_{s,k}, t_{s,1:k}) \exp(2\Delta_{s,k} + r_{s,k}|\lambda/2) \exp(-r_{s,k}|\lambda/2)
\]
\[
= Z_{s,k,C}^{-1} GIG(2\Delta_k + r_k|\bar{\lambda}, \varepsilon_{k-1,C}, \lambda),
\]  
(6.6)
where \(\lambda = (p - k + 1)(p - k)/2, \bar{\lambda} = 1 - p/2, \varepsilon_{s,k-1,C} = (m_{s,c1} - m_{s,c2})(m_{s,c1} - m_{s,c2})',\)
\(C = \{c_1, c_2\} \in \pi_{s,k-1}\), and \(GIG(\lambda, \varepsilon, \psi)\) is the generalized inverse Gaussian with parameters \(\{\lambda, \varepsilon, \psi\}\). This leads to
\[
Z_{k,C} \propto K_{\bar{\lambda}}(\sqrt{\varepsilon_{s,k-1,C}}) \exp(\lambda r_{s,k}/2)/(\lambda \varepsilon_{s,k-1,C})^{\bar{\lambda}/2},
\]  
(6.7)
where \(K_{\bar{\lambda}}(z)\) is the modified Bessel function of second kind. Hence, the conditional posterior distributions for SMC sampling are obtained as
\[
\Delta_{k} | \pi_{s,k}, t_{s,1:k-1} \sim GIG(2\Delta_{s,k} + r_{s,k}|\bar{\lambda}, \varepsilon_{k-1,C}, \lambda),
\]
\[
C^{*} | \pi_{s,k-1}, t_{s,1:k-1} \sim Multinomial(C^{*} | w_{s,k-1})
\]
where \(w_{s,k-1}\) is the vector of normalized weights, ranging over all pairs and computed using equation (6.7), and \(C^{*}\) is the pair of \(\pi_{s,k-1}\) that gets merged in \(\pi_{s,k}\).

6.3.4 Update mean locations

The posterior density for \(\mu_s\) is
\[
p(\mu_{s} | \ldots) \propto N(\mu_{s} | m, \tau I_{p \times p}) Z_s(x_{i,u_i=s} - \mu_s | \pi_{s,p-1}, t_{1:p-1}) \prod_{k=1}^{p-1} Z_{s,k}(x_{i,u_i=s} - \mu_s | \pi_s, t_s).
\]
Metropolis-Hastings sampling is used to update the mean locations. To avoid slow mixing and a high rejection rate, a customized Metropolis-Hastings is developed. The idea is to use a Gaussian distribution to locally approximate the conditional
posterior at each iterate. As the likelihood (6.2) is a product of normal density functions, the posterior of $\mu$ is expected to be not dramatically different from Gaussian.

The proposal distribution then incorporates the posterior information by applying Newton’s optimization method involving the gradient and Hessian of the posterior distribution. The focus is thus in finding MAP estimates of the mean locations instead of exploring their posterior distributions. This method was first proposed by Qi and Minka (2002). More specifically, for cluster $s$, denote $\mu^{t-1}_s$ as the parameter drawn at the $t$th MCMC iteration. Let $g(\mu^{t-1}_s) = d \log p(\mu_s|\ldots)/d\mu_s$, with the derivative evaluated at $\mu_s = \mu^{t-1}_s$, and $H(\mu^{t-1}_s) = d^2 \log p(\mu_s|\ldots)/d\mu_d\mu'$; thus, $g(\mu^{t-1}_s)$ and $H(\mu^{t-1}_s)$ are the gradient and Hessian for the log posterior, respectively. To direct the proposal distribution to a higher posterior density region, the mean of the proposal distribution uses a one-step Newton’s search

$$mm_s = \mu^{t-1}_s - h g(\mu^{t-1}_s) H(\mu^{t-1}_s)^{-1},$$

where $h$ is the “learning” rate. If the posterior distribution is far from Gaussian, a small random $h$ will avoid Markov chains “sticking” in local regions. The variance of the proposal distribution is then $-H(mm_s)^{-1}$ as a result of using a Taylor series approximation of the posterior at $\mu_s = mm_s$. Thus, the proposal distribution is $q(\mu^*_s|\mu^{t-1}_s,\ldots) = N(mm_s, -H(mm_s)^{-1})$. Since the exact form of the posterior distribution of $\mu_s$ depends on the tree structure, both the gradient and the Hessian are obtained numerically using a complex step differentiation algorithm (Skrainka, 2009). In addition, $-H(mm_s)^{-1}$ may not always be a positive definite matrix; in such a case, the modified Cholesky algorithm is used. Hence, $\mu^*_s$ is accepted with probability

$$\min \left\{ \frac{p(\mu^*_s|\ldots)N(\mu^{t-1}_s mm^*_s, -H(mm^*_s)^{-1})}{p(\mu^{t-1}_s|\ldots)N(\mu^*_s|mm_s, -H(mm_s)^{-1})}, 1 \right\},$$

where $mm^*_s = \mu^*_s - h g(\mu^*_s) H(\mu^*_s)^{-1}$. If $\mu^*_s$ is accepted, then $\mu^*_s = \mu^*_s$; otherwise, $\mu^*_s = \mu^{t-1}_s$. 110
The above sampling scheme updates all the elements of $\mu_s$ simultaneously. This procedure is adequate under moderate dimensions. With increasing dimension, a “block-at-a-time” algorithm (Chib and Greenberg, 1995) often enhances the acceptance rates in Metropolis-Hastings. Chib and Ramamurthy (2010) proposed a tailored randomized block Metropolis-Hastings method that randomly groups the parameters into an arbitrary number of blocks at every MCMC iteration for sampling dynamic stochastic general equilibrium models. Within each block, a tailored proposal density is obtained. Similar to this approach, $\mu_s$ is randomly grouped under high dimensional data, and groups are sequentially updated using Metropolis-Hastings.

In addition, to facilitate the jumping of the Markov chains to the mode of the posterior, the steps of sampling tree structures and mean vectors are looped a small number of times before sampling the other parameters within each iteration. Across multiple numerical experiments, I have also found it useful to initialize the MCMC by numerically finding MAP estimates of $\mu$, i.e. by using the MCMC as described but, for a few hundred initial iterations, simply obtain MAP estimates of $\mu$. This has been found to be very beneficial in moving into the region of the posterior, especially under higher dimensions, and then running the full MCMC thereafter.

6.4 Simulation studies

To assess the usefulness and efficacy of the developed mixture of binary trees model and the customized MCMC algorithms, several simulation studies are conducted in this section.

*Example 1.* A sample of size $n = 500$ was drawn from a $p = 5$–dimensional mixture of 2 Gaussian components with equal proportions, mean vectors of $(3, 3, 2, 0, 0)'$
and \((3, 1, 0, 3, 3)'\) and covariance matrices
\[
\begin{pmatrix}
2 & 1 & 0.8 & 0 & 0 \\
1 & 2 & 0.8 & 0 & 0 \\
0.8 & 0.8 & 2 & 0 & 0 \\
0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 2
\end{pmatrix}
\quad \begin{pmatrix}
2 & -0.6 & 0 & 0 & 0 \\
-0.6 & 2 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 \\
0 & 0 & 0 & 2 & 0.5 \\
0 & 0 & 0 & 0.5 & 2
\end{pmatrix}
\]

Figure 6.4: Pairwise scatter plots of the generated data set in Example 1.

Figure 6.4 shows a scatter plot of the generated sample. The model specification used \(S = 6\) for the maximum allowed number of components. The number of particles \(M\) in the SMC algorithm is set to be 100. The learning rate \(h\) is set as 1. The other parameters are set as \(e = 2\), \(f = 1\), \(m = 0_{5\times1}\), \(\tau = 3\), and \(s_z = 0.5\). The MCMC computations are initialized based on the prior distributions. 10,000 iterations are computed. Based on the parameter values in the last iteration, the
model identifies two non-empty components. The trace plots in Figure 6.5 provide a global visualization addressing MCMC convergence and the Hessian-based proposal distribution improves the acceptance rate.

![Figure 6.5: Trace plot for $\mu_{s,j}$ under the last 1000 iterations, $s = 1, 2$, $j = 1 : 5$](image)

The estimated tree structures are provided in Figure 6.6. As discussed in Section 6.2.2, the time at which a pair of nodes merge indicates how the nodes are related. Under the first component, the first three variables are positively correlated; therefore, they merged first. The remaining variables are independent, and they merged last. Under the second component, the last two variables are the only positively correlated pair; therefore, they merged first. The first two variables are negatively correlated; because the second variable is merged first, the first variable must merge last. The clustering result are illustrated in Figure 6.7; this suggests that the mixture of trees model can also detect Gaussian distributed subpopulations.

Figure 6.8 provides a more detailed examination on the behavior of the binary trees from the MCMC algorithm. Each row in the plot represents one MCMC output, columns represent components. The left column indicates that the tree structures are preserved, though with varying branching times. The right column shows that the variable 4 and 5 are always merged first and very fast as they are the only
positively correlated pair. Conditioning on the next merged pair, the tree structures are preserved in the sense that the negatively correlated pair will merge last. In addition, compared with the first merging time, the second merge takes a relatively longer time as the tree is now merging uncorrelated and/or negatively correlated nodes. Based on this observation, a more sophisticated prior on the binary tree might be suggested, one that extends the Kingman’s coalescent by allowing optional stopping of the merging process of a subset of the tree. Table 6.1, which shows the classification results compared with the truth based on the 4 consecutive MCMC outputs, additionally demonstrates that the clustering performance depends largely on the tree’s structure, hence it is tempting to marginalize out the time $t$ to obtain a marginal posterior distribution on the structure of binary tree $\pi$ only to facilitate MCMC sampling and posterior inference.

Example 2. A sample of size $n = 1000$ was drawn from a $p = 10$—dimensional mixture of 5 Gaussian components with equal proportions, mean vectors of $(1.5, 1.5, 0_{1\times8})'$, $(0, 1.5, 1.5, 0_{1\times7})'$, $(0_{1\times2}, 1.5, 1.5, 0_{1\times6})'$, $(0_{1\times3}, 1.5, 1.5, 0_{1\times5})'$, and $(0_{1\times4}, 1.5, 1.5, 0_{1\times4})'$. 

Figure 6.6: The estimated tree structures from the last MCMC iteration in Example 1. Color edges are for easy visualization.
Figure 6.7: Pairwise scatter plots of the generated data set in Example 1. Colors serve as component indicators obtained based on the last MCMC iteration.

Each component has equal variance of 1, and variables with non-zero mean are positively correlated with magnitude 0.7, so that the last four dimensions are of the same distribution across all the components and only the first 6 dimensions are discriminating. Figure 6.9 shows a scatter plot of the generated sample; no clear structures can be inferred from the pair-wise plots alone.

Table 6.1: Example 1. Classification rate for the 2 estimated mixture components based on 4 consecutive MCMC outputs corresponding to Figure 6.8. The classification rate is obtained by dividing the true positive by the true size of that component.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Component 1</th>
<th>Component 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration 1</td>
<td>0.8455</td>
<td>0.9606</td>
</tr>
<tr>
<td>Iteration 2</td>
<td>0.8537</td>
<td>0.9606</td>
</tr>
<tr>
<td>Iteration 3</td>
<td>0.8902</td>
<td>0.9567</td>
</tr>
<tr>
<td>Iteration 4</td>
<td>0.8821</td>
<td>0.9567</td>
</tr>
</tbody>
</table>
Figure 6.8: The estimated tree structures from four consecutive MCMC iterations in Example 1. Each row represents one MCMC iteration and columns represent two components.

I first fit the sample with the standard DP Gaussian mixture model. Analysis allows up to 16 components using default, relatively vague priors, so encouraging
smaller components. Using the standard MCMC technique for posterior simulation, a posterior sample of size 50,000 was generated, initialized at the random starting points generated from the priors. The same procedure is repeated several times. Each fitted model has only one dominant Gaussian component with proportion greater than 0.9. This result indicates that the DPM model is not sensitive under relatively small data set. It simply mixes all the data together under one distribution, masking the finer subpopulation structure. The approximate posterior mean of that dominant component is \( (0.28, 0.66, 0.61, 0.51, 0.64, 0.30, -0.01, 0.05, 0.01, -0.02)' \).

I contrast the above with results from analysis using the new mixture of binary trees model. Model specification uses \( S = 10 \) components. The other parameters are set as \( M = 100, h = 1, e = 10, f = 1, m = 0_{10 \times 1}, \tau = 3 \) and \( s_z = 0.01 \). The MCMC
computations are initialized based on the prior distributions. 10,000 iterations are computed. The same procedure is repeated several times. Under each fitted model, based on the parameter estimation in the last iteration, the model identifies more than 5 non-empty components; 5 corresponds to the 5 “true” components of interest, while the rest of the non-empty components are of very small size, just accommodating outliers with respect to the 5 main components. The estimated tree structures corresponding to the 5 main components are shown in Figure 6.10. The synthetic covariance structures are adequately captured by the binary trees.

![Figure 6.10: The estimated tree structures in Example 2 based on the last MCMC iteration](image)

**Example 3.** A sample of size $n = 1000$ was drawn from a $p = 10$–dimensional mixture of 4 Gaussian components with equal proportions, mean vectors of $(3, 3, 3, 0_{1 \times 5}, 1.5, 1.5)^	op$, $(3, 3, 3, 0, 1.5, 1.5, 0_{1 \times 4})^	op$, $(1.5, 1.5, 0_{1 \times 6}, 1.5, 1.5)^	op$, and $(1.5, 1.5, 0_{1 \times 2}, 1.5, 1.5, 0_{1 \times 4})^	op$. The covariance structure for the first three dimensions in the first two com-
ponents is
\[
\begin{pmatrix}
1.5 & 0.7 & 0.7 \\
0.7 & 1.5 & 0.7 \\
0.7 & 0.7 & 1.5
\end{pmatrix},
\]
and in each component the covariance structure for dimensions having mean 1.5 is
\[
\begin{pmatrix}
1 & 0.7 \\
0.7 & 1
\end{pmatrix}.
\]
Variables in the remaining dimensions are uncorrelated with variance 1.5 in the first three components and 1 in the last component. Hence, the number of discriminating dimensions is reduced compared with Example 2; e.g., only four dimensions distinguish component 1 and 2. Figure 6.11 shows a scatter plot of the generated sample; a two-cluster structure can be visually identified.

I first fit the sample with the standard DP Gaussian mixture model. Analysis allows up to 16 components using default, relatively vague priors, so encouraging smaller components. Using the standard MCMC technique for posterior simulation, a posterior sample of size 50,000 was generated, initialized at the random starting points generated from the priors. The same procedure is repeated several times. Similar to Example 2, the DPM model clusters a large portion of the data into one group, clearly masking the true structure.

Next, the data set is fitted with hierarchical clustering with coalescents using the SMC algorithm developed in Henao and Lucas (2012), i.e. a binary tree is built with 1000 leaf nodes representing observations. The prior specification for modeling covariance among 10 dimensions uses the inverse Wishart distribution. Both greedy algorithm and the SMC using 100 particles and 200 iterations are performed. The estimated binary trees are cut allowing 7 clusters, 4 of which identify the main “true” structure. The other identified clusters have observations. Table 6.2 shows the classification results compared with the truth using the greedy algorithm. The
Figure 6.11: Pairwise scatter plots of the first 5 dimensions of the generated data set in Example 3.

estimated component 1 is the largest component; it groups data generated from all the 4 normal components. Table 6.3 shows the classification results using the SMC sampling. The classification rate is slightly improves, however, the true positive rates are only around 50% for all the 4 components. In addition, there is no automatic method on where to cut the tree.

I contrast the above with results from analysis using the mixture of binary trees model. Model specification uses \( S = 10 \) components. The other parameters are set as \( M = 100, \ h = 1, \ e = 10, \ f = 1, \ m = 0_{10 \times 1}, \ \tau = 3 \) and \( s_z = 0.01 \). The MCMC computations are initialized based on the prior distributions. 10,000 iterations are computed. The same procedure is repeated several times. Under each fitted model,
Table 6.2: Example 3. Classification rate for 4 major estimated components resulting from the hierarchical clustering procedure. Comp. \( j \) represents the normal component \( j \) and Estimated Comp. \( j \) is the estimated component \( j \). The classification rate is obtained by dividing the true positive by the true size of that component.

<table>
<thead>
<tr>
<th></th>
<th>Comp. 1</th>
<th>Comp. 2</th>
<th>Comp. 3</th>
<th>Comp. 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated Comp. 1</td>
<td>0.3465</td>
<td>0.6695</td>
<td>0.1547</td>
<td>0.5143</td>
</tr>
<tr>
<td>Estimated Comp. 2</td>
<td>0.2323</td>
<td>0.0932</td>
<td>0.1811</td>
<td>0.0245</td>
</tr>
<tr>
<td>Estimated Comp. 3</td>
<td>0.4173</td>
<td>0.1441</td>
<td>0.4075</td>
<td>0.0245</td>
</tr>
<tr>
<td>Estimated Comp. 4</td>
<td>0.0039</td>
<td>0.0424</td>
<td>0.2566</td>
<td>0.3592</td>
</tr>
</tbody>
</table>

Table 6.3: Example 3. Classification rate for 4 major estimated components resulting from the hierarchical clustering procedure. Comp. \( j \) represents the normal component \( j \) and Estimated Comp. \( j \) is the estimated component \( j \). The classification rate is obtained by dividing the true positive by the true size of that component.

<table>
<thead>
<tr>
<th></th>
<th>Comp. 1</th>
<th>Comp. 2</th>
<th>Comp. 3</th>
<th>Comp. 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated Comp. 1</td>
<td>0.5236</td>
<td>0.1695</td>
<td>0.2906</td>
<td>0.0898</td>
</tr>
<tr>
<td>Estimated Comp. 2</td>
<td>0.0984</td>
<td>0.5720</td>
<td>0.0151</td>
<td>0.1184</td>
</tr>
<tr>
<td>Estimated Comp. 3</td>
<td>0.1024</td>
<td>0.0127</td>
<td>0.5509</td>
<td>0.2082</td>
</tr>
<tr>
<td>Estimated Comp. 4</td>
<td>0.0079</td>
<td>0.0720</td>
<td>0.1208</td>
<td>0.5633</td>
</tr>
</tbody>
</table>

Based on the parameter estimation in the last iteration, the model identifies a few more than 4 non-empty components; 4 of these correspond to the 4 true in every run. The estimated tree structures corresponding to the 4 main components are shown in Figure 6.12; this shows that the synthetic covariance structures are adequately captured by the binary trees.

A classification table is also computed. Table 6.4 shows the classification results from the mixture of tree model. The classification rate is dramatically improved such that the majority of the data are correctly assigned to their true components.

6.5 Discussion

In this chapter, I have defined and explored a specific class of mixture models, the mixture of binary trees, with the goal of exploring data structures from low dimensional subspaces. The approach utilizes binary trees as a tool for modeling
Figure 6.12: The estimated tree structures in Example 3 based on the last MCMC iteration.

Multivariate data. The binary trees are flexible enough to capture the dependencies between variables represented by the means of graphs. The simulation studies compare with DPM models and demonstrate that the new model can effectively capture the information concerning the underlying distribution with a relatively small number of parameters. The model will continue to be tested on high-throughput, high-dimensional flow cytometry data. The current posterior inferences are based on the parameter values from one MCMC iteration because of the lack of interpretation on the average of binary trees. It is tempting to develop a more sophisticated posterior inference scheme utilizing information from all the converged MCMC iterations. Furthermore, a Bayesian EM algorithm and/or variational Bayesian algorithm might
Table 6.4: Example 3. Classification rate for 4 major estimated components resulting from the mixture of trees model. Comp. $j$ represents the normal component $j$ and Estimated Comp. $j$ is the estimated component $j$. The classification rate is obtained by dividing the true positive by the true size of that component.

<table>
<thead>
<tr>
<th>Estimated Comp.</th>
<th>Comp. 1</th>
<th>Comp. 2</th>
<th>Comp. 3</th>
<th>Comp. 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated Comp. 1</td>
<td>0.7441</td>
<td>0.1568</td>
<td>0.1208</td>
<td>0</td>
</tr>
<tr>
<td>Estimated Comp. 2</td>
<td>0.0984</td>
<td>0.6271</td>
<td>0</td>
<td>0.0122</td>
</tr>
<tr>
<td>Estimated Comp. 3</td>
<td>0.1181</td>
<td>0.0042</td>
<td>0.7019</td>
<td>0.0286</td>
</tr>
<tr>
<td>Estimated Comp. 4</td>
<td>0.0394</td>
<td>0.1992</td>
<td>0.1698</td>
<td>0.9224</td>
</tr>
</tbody>
</table>

be developed to find MAP estimates.

The idea of using binary tree structures to model variable dependencies is a fundamental element in the construction of more sophisticated hierarchical models. In addition to clustering, many problems involve comparing clusters across multiple samples. The hierarchical version of mixture of binary tree models can be used to permit information sharing among data sets.
Summary and future work

This dissertation has focused on a range of related problems in Bayesian mixture model development for classification, design and variable selection, structured and hierarchical non-parametric Bayesian methods, rare event detection, and statistical computation involving simulation and optimization. Much of this is linked to problems involving large data sets, with originating motivations in analysis of biomedical data from flow cytometry technologies. Chapter 2 presents a new Bayesian EM algorithm in fitting DPM models and a new way to define clusters in mixture models. Data sets of increasing scale and complexity pose challenges to standard statistical methods, and these are exemplified in areas where one goal is classification and discrimination of subpopulations. Chapter 3 presents a new method in identifying subsets of variables that play roles in discrimination of subpopulations in the context of multivariate mixture modeling, developed and applied to these general variable selection/design questions in flow cytometry applications. Chapter 4 illustrates that the method is both effective and computationally attractive for routine use in assessing and prioritizing subsets of variables according to their roles in discriminating subpopulation structure in the analysis of very large cytometry data sets. Classifica-
tion and cluster analysis methodologies often do not scale well with data dimension. As the number of measured variables grows, there is an increasing need to consider structured, hierarchical models to enable sensitive inference on subpopulation structure. Moreover, as sample sizes increase we often face problems of masking of subtler substructure; model fitting can often lack the ability to identify “rare events” due to the dominance of much of the data. In use of mixture models, one general technique for addressing these issues is to encourage a sparse mixture model structure. Chapter 5 introduces novel, hierarchical nonparametric Bayesian mixture models that address both problems in a specific biomedical context. Chapter 6 further extends the model to explore data structure from low dimensions using tree-based concepts.

There are many exciting directions for future research. It would be interesting to further develop variable selection methods and their interfaces with Bayesian decision theory. One specific direction I am exploring is to incorporate or modify the developed discriminative information measure into hierarchical priors, such that a priori the measured variables can be of varying relevance to the classification problem. I also aim to expand applications of new mixture models and methods in other areas, both in biomedical work and flow cytometry, and in new areas of mixture-based classification. The theme of variable selection is also central in temporal studies, and I plan to expand my research into problems requiring new ways of approaching prioritization and selection of variables in multivariate time series analysis. I would also like to extend my current hierarchical mixture models to sequentially partition data on each univariate dimension in increasing large-scale problems coupled with GPU computation to enable fast Bayesian analysis, and also develop systematic approaches to integrating contextual information into the selection of orderings of variable subsets to define the hierarchies.

One of my recent application of the hierarchical mixture modeling is to rare event detection and cell subset alignment across flow cytometry samples. Various DPM
model based clustering methods, including the models developed in this dissertation, apply to data samples independently. Given multiple data samples, two questions are often of interest: (i) to align cluster labels across data sets, such that comparing clusters across multiple samples is possible, which is usually the purpose of the original experiments. (ii) identify very rare event clusters due to *masking* by abundant event clusters under each sample. Both questions can be successfully addressed by the use of hierarchical version of the DPM models. Such hierarchical, or multi-level models, represent individual observation in a data set as being organized into successively higher units. For example, individual observations belong to a sample, and a sample may belong to a batch of samples. The critical idea is that data subset characteristics that are common across data samples can be used to inform and hence better characterize observations in individual samples. For example, by placing all data samples under a common prior, such that the location and the shape in any of the individual sample components are shared across all samples, but the weight (proportions) of the component in each sample is unique. This hierarchical model leaves the cluster locations and shapes constant across data sets, and hence aligns the clusters in that the location of the components is common to all data sample. The hierarchical DPM models also allow information sharing over data sets. In other words, if a rare cluster is found in more than one of the samples, this information is shared across the data sets to detect the cluster even though the frequency in a particular data set may be vanishingly small. This hierarchical model thus increases sensitivity for clustering subtypes that are of extremely low frequency in one sample but common to many samples or present in high frequency in one or more samples. In principle, there is no lower limit to the size of a cluster that can be detected in a particular sample, so long the algorithm is able to “borrow strength” from other samples as described. In practice in flow cytometry, vanishingly small clusters (e.g. 3 – 5 events out of 100,000) require expert interpretation to distinguish background
from signal, but it is not uncommon for biologically significant antigen-specific cells to be present at such frequencies. Several experimental studies have shown that the hierarchical model is preferable to other methods such as using a reference data sample or pooling the data from all samples, since individual sample characteristics are lost with these alternative strategies.
Bibliography


131


Biography

Lin Lin was born in Nanjing, Jiangsu, China. She attended National University of Singapore for her undergraduate degree in Singapore. She graduated with her Bachelors of Science in Statistics in May 2007. In July 2008, she obtained her Masters of science (thesis) in Statistics from the same university. In August 2008, she enrolled as a graduate student in the statistical science department at Duke University in Durham, North Carolina. She was advised by Dr. Mike West. Her doctoral research focused on Bayesian statistics, with a particular interest in nonparametric Bayesian methodology.