BAYESIAN MODEL-BASED APPROACHES WITH MCMC COMPUTATION TO SOME BIOINFORMATICS PROBLEMS

A Dissertation

by

KYOUNGHWA BAE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2005

Major Subject: Statistics

BAYESIAN MODEL-BASED APPROACHES WITH MCMC COMPUTATION TO SOME BIOINFORMATICS PROBLEMS

A Dissertation

by

KYOUNGHWA BAE

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Bani K. Mallick (Co-Chair of Committee)	Christine G. Elsik (Co-Chair of Committee)
Thomas R. Ioerger (Member)	Paul F. Dahm (Member)

Michael T. Longnecker
(Head of Department)

May 2005

Major Subject: Statistics

ABSTRACT

Bayesian Model-based Approaches with MCMC Computation to Some

Bioinformatics Problems . (May 2005)

Kyounghwa Bae, B.S., Korea University;

M.S., Korea University

Co-Chairs of Advisory Committee: Dr. Bani K. Mallick
Dr. Christine G. Elsik

Bioinformatics applications can address the transfer of information at several stages of the central dogma of molecular biology, including transcription and translation. This dissertation focuses on using Bayesian models to interpret biological data in bioinformatics, using Markov chain Monte Carlo (MCMC) for the inference method. First, we use our approach to interpret data at the transcription level. We propose a two-level hierarchical Bayesian model for variable selection on cDNA Microarray data. cDNA Microarray quantifies mRNA levels of a gene simultaneously so has thousands of genes in one sample. By observing the expression patterns of genes under various treatment conditions, important clues about gene function can be obtained. We consider a multivariate Bayesian regression model and assign priors that favor sparseness in terms of number of variables (genes) used. We introduce the use of different priors to promote different degrees of sparseness using a unified two-level hierarchical Bayesian model. Second, we apply our method to a problem related to the translation level. We develop hidden Markov models to model linker/non-linker sequence regions in a protein sequence. We use a linker index to exploit differences

in amino acid composition between regions from sequence information alone. A goal of protein structure prediction is to take an amino acid sequence (represented as a sequence of letters) and predict its tertiary structure. The identification of linker regions in a protein sequence is valuable in predicting the three-dimensional structure. Because of the complexities of both models encountered in practice, we employ the Markov chain Monte Carlo method (MCMC), particularly Gibbs sampling (Gelfand and Smith, 1990) for the inference of the parameter estimation.

To my parents, my sister, my brother

and

My love Jason

ACKNOWLEDGEMENTS

I would like to express my gratitude to Dr. Bani K. Mallick, my advisor, for his kind support and encouragement throughout the years. I am also deeply grateful to my co-advisor, Dr. Christine G. Elsik, for her guidance and support for my challenging research topic. It would have been impossible without her guidance and support. I would also like to extend my thanks to Dr. Ioerger and Dr. Dahm for their suggestions and help. I would like to give a special thanks to Dr. Longnecker for his generosity and his support throughout my study and life in Texas A&M Univesity. There are many people who have made important contributions to this dissertation. I want to thank you to all of you who gave me lots of support and love. I wish to thank specially Sinae Kim, Daisy Liu, Dukjin Nam, Kyongryun Kim and Deukwoo Kwon for their help and friendship. I am so glad to have met them and become friends with them during my study at Texas A&M University. I also would like to thanks to Kyoungshim for listening to all the complaints and giving me a great deal of support. I cannot express my thanks enough to my parents, my sister and my brother for their love, endless encouragement and concern. I would like to thank Valentine and Robert who are like my parents in this country and Nicole who is like my sister. To all people, I am most thankful to Jason, my love, who gives me love and strength to get through all the hard time and for his support during this long journey.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Motivations and Problems

The explosion of interest in bioinformatics has been driven by the emergence of experimental techniques that generate data in a high throughput fashion - such as DNA sequencing, mass spectrometry, and microarray expression analysis (Miranker, 2000; Altman and Raychaudhuri, 2001). Bioinformatics arose from the availability of large data sets that are too complex to allow manual analysis. High-throughput genomic approaches are generating vast amounts of DNA and protein sequence and structure data, genetic map information, and gene expression profiles. The pace of data accumulation is rapidly outrunning the rate of data processing and comprehension. We have an ever-growing amount of biological data with advances in microarray technologies and the genome projects of various species. Implementing appropriate statistical models to interpret this data is a difficult and important problem in bioinoformatics. Bayesian data analysis is a method which enable us to make inferences from data using probability models for quantities we observe and for quantities about which we wish to learn. The essential characteristic of Bayesian methods is their explicit use of probabilities for quantifying uncertainty in inferences based on statistical data analysis. We have developed Bayesian models which provide probabilities for quantifying uncertainty in solving some problems in bioinformatics. A common method for determining the level of gene expression is to measure the amount of mRNA being

---

This dissertation follows the style of *Biometrics*.

produced by that gene. A microarray experiment consists of mRNA extracted from cells under different treatment conditions and glass slides (the microarrays) to which spots of genetic material are attached. This microarray data has few samples but a large number of genes at each sample. The most commonly used computational method for analyzing microarray data is to filter or reduce the data and to apply classification or clustering methods to the reduced data. These methods provide a compact summarization of the data and point to functional relationships between clustered genes. However, classification and clustering methods fail to highlight or rank the most important few genes among thousands of genes. Variable selection in cDNA microarray data highlights those genes which exhibit a different gene expression between two tissue types (e.g. normal and cancer) by removing redundant variables. We develop a two-level hierarchical Bayesian model for variable selection on cDNA data. We consider a multivariate Bayesian regression model and assign priors that favor sparseness in terms of number of variables (genes) used. We introduce the use of different priors to promote different degrees of sparseness using a unified two-level hierarchical Bayesian model.

Besides gene expression data analysis, the study of protein structure, folding and function has a long history. Protein folding and its role in determining function from sequence information has been an intensive subject of research. How to predict the three dimensional structure of a protein from its amino acid sequence is the major unsolved problem in structural molecular biology (Branden and Tooze, 1999). It requires enormous amounts of computing time in addition to the complication that the energy difference between a stable folded molecule and its unfolded state is a small number containing large errors. The use of statistical methods for protein structure prediction is a natural approach. The common thread shared by these approaches is to attempt to use the existing database of experimentally determined structures in

order to infer structure for new sequences. Statistical methods broadly encompass techniques such as sequence homology search followed by structural homology modeling, threading and fold recognition, secondary structure prediction, and a variety of other methods which at some level involve fitting statistical models to a database structure (Schmidler, Liu, and Brutlag, 2001).

The methodology developed in this dissertation is related to the particular protein structure prediction, which is intermediate step to the three-dimensional structure prediction. The functional properties of proteins depends on their three-dimensional structures. The three-dimensional structure arises because particular sequences of amino acids in polypeptide chains fold to generate, from linear chains, compact domains with specific three-dimensional structures (Branden and Tooze, 1999). To understand the biological function of proteins we would like to predict the three-dimensional structure from the amino acid sequence. We wish to extract principles of protein sequence and structure from the database that may be used to accurately predict the structure of novel sequences.

Hidden Markov models (HMMs) have been employed in diverse areas of computational biology: genetic linkage maps (Lander and Green, 1987), multiple alignment of protein families (Haussler, Krogh, Mian, and Sjolander, 1993), gene prediction (Kulp, Haussler, Reese, and Eeckman, 1996; Burge and Karlin, 1997; Henderson, Salzberg, and Fasman, 1997), the secondary structure prediction (Schmidler, Liu, and Brutlag, 2000; Schmidler et al., 2001), etc. The observations in these HMMs for protein structure prediction are recognized as strings of amino acids (categorical variables), forming the primary sequence of a protein.

We focus on using Bayesian models to predict the linker regions in a protein sequence from sequence information alone and use Markov chain Monte Carlo (MCMC) for the inference method. The *domain* is the fundamental functional and three-

dimensional structural unit of protein structure. Many structural domains evolve as independent units that are found in different combinations. Thus, the domain has alternatively been defined as an evolutionary unit. The identification of domains within a protein sequence is valuable in numerous applications. An alternative to delineating domain boundaries is identifying inter-domain linker boundaries. The *linker* is defined as a region between adjacent domains. We assume that protein sequence data is produced by a hidden Markov model and compositional variation is likely to reflect functional or structural differences between regions. We develop hidden Markov models (HMMs) to model linker/non-linker sequence regions using a linker index to exploit differences in amino acid composition between regions.

Because of the complexities of both models encountered in practice, we employ a Markov chain Monte Carlo method (MCMC), particularly Gibbs sampling (Gelfand and Smith, 1990) for the inference of the parameter estimation. Gibbs sampling effectively reduces the problem of sampling from a high-dimensional distribution to sampling from a series of low-dimensional distributions.

1.2 Bayesian Data Analysis and Markov Chain Simulation

1.2.1 Fundamentals of Bayesian Analysis

Bayesian inference is the process of fitting a probability model to a set of data and summarizing the result by a probability distribution on the parameters of the model and on unobserved quantities such as predictions for new observations. This is a different approach from the classical ones, in which the parameters of the model are estimated using the distribution of data values y conditional on the true unknown values of θ. The most important characteristic of Bayesian methods is their explicit use of probability for quantifying uncertainty in inferences based on statistical data analysis. The process of Bayesian data analysis can be divided into the three steps. First,

we set up a full probability model, a joint probability distribution for all observable and unobservable quantities in a problem. Second, conditioning on observed data, we calculate and interpret the appropriate *conditional posterior distribution*, which is the conditional probability distribution of the unobserved quantities of interest given the observed data. Third, we evaluate the fit of the model and the implications of the resulting posterior distribution (Gelman, Carlin, Stern, and Rubin, 2000). Throughout this dissertation, we use the term 'the conditional posterior distribution' and 'the posterior distribution' interchangeably. Bayesian statistical conclusions about parameters θ are made in terms of probability statements conditioning on observed data y, $P(\theta|y)$. The core of Bayesian inference is to develop a model which has *a joint probability distribution* $P(\theta, y)$ and perform the necessary computations to summarize $P(\theta|y)$. In order to make probability statements about θ given y, we begin with model providing a joint probability distribution y and θ. The joint probability distribution y and θ can be expressed as a product of *prior distribution* (the probability distribution of the parameters) $p(\theta)$ and the *likelihood distribution,* $p(y|\theta)$ as follows.

$$P(\theta, y) \propto p(\theta) \times p(y|\theta)$$

It is an important feature of Bayesian inference to incorporate the the expert opinions, historical data etc. through the prior distribution. If not fixed at particular numerical values, the parameters of the prior distributions are called *hyper-parameters.* Conditioning on data y, the posterior distribution of parameters θ, $p(\theta|y)$ is proportional to the product of the likelihood distribution and the prior distribution.

$$\begin{aligned} P(\theta|y) &= \frac{p(\theta, y)}{p(y)} = \frac{p(\theta) \times p(y|\theta)}{\int p(\theta) \times p(y|\theta) d\theta} \\ &\propto p(\theta) \times p(y|\theta) \end{aligned}$$

Conjugacy is the property that the posterior distribution follows the same parametric form as the prior distribution. Another important feature of Bayesian inference is that

the posterior distribution is centered at a point that represents a compromise between the prior information and the data, and the compromise increasingly is controlled by the data as the sample size increases. In the case that conclusions will be drawn about one or only a few parameters at a time, we just need to obtain *the marginal posterior distribution* of the particular parameters of interest. We obtain the marginal posterior distribution by first obtaining the joint posterior distribution of all unknowns and then integrate this over the unknowns that are not of interest.

It is almost impossible to usefully characterize posterior distributions analytically because they can be very complex in a high dimensional space. However a sample of points drawn from such a distributions can provide a satisfactory picture of it. In particular, from such a sample we can obtain *Monte Carlo* estimates for the expectations of various random functions of the variables. The Monte Carlo method approximates a solution by introducing a random vector U that is uniformly distributed on the region of integration. Applying function f to U , the Monte Carlo estimator of the expectation of $f(U)$, denoted $E[f(U)]$, is as follows.

$$\int f(u)du \approx \frac{1}{m}\sum_{t=1}^{m} f(U^t)$$

where m is the number of drawn points from $f(U)$.

1.2.2 Markov Chain Monte Carlo

The posterior distributions are usually in a complicated form so it is hard to generate samples from the posterior distribution directly. Markov Chain Monte Carlo (MCMC) techniques (Gelfand and Smith, 1990; Gilks, Richardson, and Spiegelhalter, 1996) will be used throughout this dissertation to generate samples from the posterior distributions. The key is to create a Markov process whose stationary distribution is a specified *target posterior distribution* $P(\theta|y)$ and run the simulation long enough

that the distribution of the current draws is close enough to $p(\theta|y)$. We summarize the posterior distribution and compute statistics by using these draws.

We use MCMC simulation method, specifically the Gibbs sampler, for sampling from the $P(\theta|y)$. The general procedure is as follows.

- Using a starting point, run independent parallel sequences of an iterative simulation, such as Gibbs sampler or the Metropolis algorithm.
- Run the iterative simulation until approximate convergence appears to have been reached.
- To diminish the effect of starting distribution, we discard the beginning of the sequence (burn-in), because our inferences will be based on the assumption that the distributions of the simulated values θ^t, for large t,are close to the target distribution.
- Summarize inference about the posterior distribution by treating the set of all iterates from the simulated sequences after burn-in as an identically distributed sample from the target distribution.

The Gibbs sampler is a particular Markov chain algorithm that has been found useful in many multidimensional problems. Gibbs sampling effectively reduces the problem of sampling from a high-dimensional distribution to sampling from a series of low-dimensional distributions Suppose $\theta = (\theta_1, \cdots, \theta_d)$ and the corresponding univariate conditional distributions are $f_1, \cdots, f_d$. The distributions $f_1, \cdots, f_d$ are called *the full conditional distributions*. Also, suppose that we can simulate from these full conditional distributions.

$$\theta_{\mathbf{i}}|\theta_1, \cdots, \theta_{i-1}, \theta_{i+1}, \cdots, \theta_d \sim f_i(\theta_i|\theta_1, \cdots, \theta_{i-1}, \theta_{i+1}, \cdots, \theta_d)$$

where $i = 1, \cdots, d$. An iteration of Gibbs sampler consist of d updates of vectors in iteration t, where each update adjust one component of θ conditioning on the other $(d-1)$ components. At each iteration t, an ordering of the d subvectors of θ is chosen and, in turn, each θ_i^t is sampled from the conditional distribution given all the other components of θ. Thus each subvector θ_i is updated conditional on the latest values of θ for the other components, which are the iteration t values for the components already updated and the iteration $t-1$ values for the others.

The Gibbs Sampler Algorithm

Given $\theta^{(t)} = (\theta_1^{(t)}, \cdots, \theta_d^{(t)})$, generate

1. $\theta_1^{(t+1)} \sim f_1(\theta_1 | \theta_2^{(t)}, \cdots, \theta_d^{(t)})$
2. $\theta_2^{(t+1)} \sim f_2(\theta_2 | \theta_1^{(t+1)}, \theta_3^{(t)}, \cdots, \theta_d^{(t)})$
3. $\theta_3^{(t+1)} \sim f_3(\theta_3 | \theta_1^{(t+1)}, \theta_2^{(t+1)}, \theta_4^{(t)}, \cdots, \theta_d^{(t)})$

$\vdots$

d. $\theta_d^{(t+1)} \sim f_d(\theta_d | \theta_1^{(t+1)}, \theta_2^{(t+1)}, \cdots, \theta_{d-1}^{(t+1)})$

1.3 The Central Dogma of Molecular Biology

To provide background for the problems considered in this dissertation, we briefly review the central dogma of molecular biology. It is useful to view primary biological processes as information transfer processes. The information necessary for the functioning of cells is encoded in molecular units called genes. There are three primary information transfer processes in functioning organisms: replication, transcription and translation. These three processes, illustrated in Figure 1, make up the central dogma of molecular biology. Messages are formed from genes and these messages contain instructions for the creation (synthesis) of functional structures called proteins, which are necessary for cell life processes. The phenotypes of the cells are determined by their internal chemistry resulting from metabolic reactions. These metabolic re-

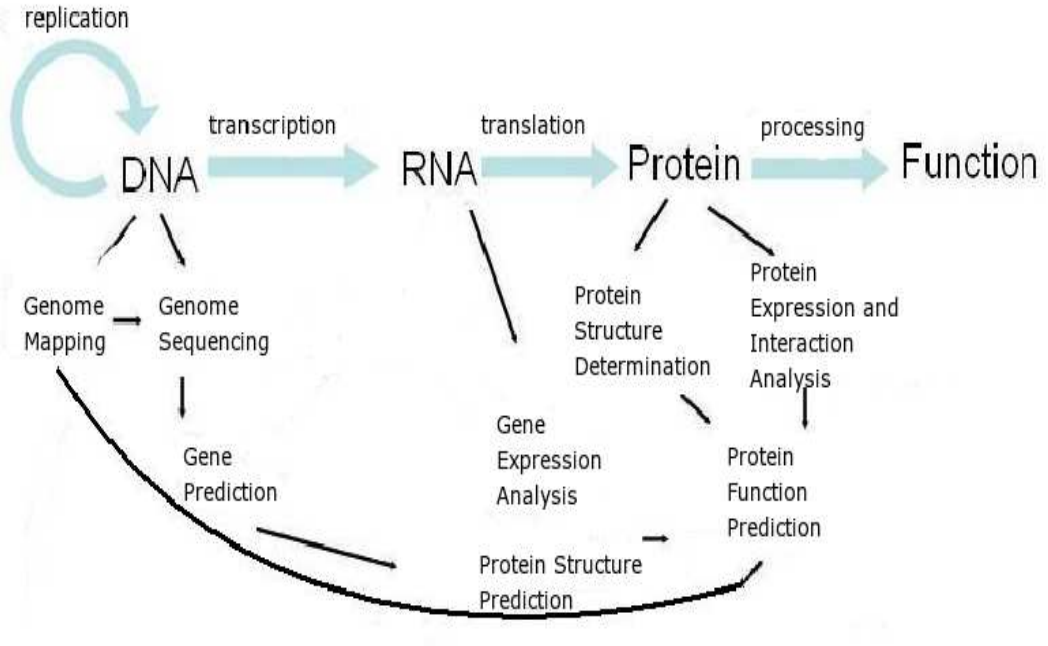

Figure 1: The central dogma of molecular biology

actions are catalyzed by a special class of proteins called enzymes. The function of a protein depends on its three-dimensional structure, which in turn depends on the linear sequence of 20 primary amino acids of the protein. The kind and amount of protein present in the cell depends on the genotype of the cell. Since genes encode the linear sequence of amino acids that form polypeptides, they specify the protein structure. Therefore, together with environmental factors, genes determine the phenotypes of cells and hence the organism. This simplified model with intermediate products, starting with genes and concluding with phenotype, is illustrated in Figure 1.

1.3.1 DNA and Microarray Experiment

We now know that with few exceptions genes are composed of deoxyribonucleic acid (DNA). DNA consists of four primary types of nucleotide molecules. The common

structure of a nucleotide contains a phosphate, a (deoxyribose) sugar and a nitrogen base. The four types of nucleotides are distinguished from one another by their distinct nitrogen base: adenine (A), guanine (G), cytosine (C) and thymine (T). Watson and Crick (1953) discovered that DNA exists as a double helix, where each helix is a chain of nucleotides. In DNA, base A pairs with T and base G pairs with C exclusively. The specific pairing of DNA bases (A-T, G-C) is called base-sequence complementarity. This complementary base pairing is the basis of a process called hybridization. DNA exists in its native state as a double helical structure. However, with sufficient heating the hydrogen bonds between complementary base pairs break and the DNA double strands separate (denature) into two single strands. Upon cooling the solution containing the DNA, the two single strands reanneal (renature) to recreate the double-stranded DNA form. Figure 1 shows how genes (DNAs) are linked to organism phenotype and illustrates the reason for measuring mRNA, the direct product of DNA transcription. DNA transcription is the information transfer process directly relevant to DNA microarray experiments, because quantification of the type and amount of this copied information is the goal of the microarray experiment. The process of transcription begins with DNA in the nucleus where the DNA template strand is copied. The copied strand is called messenger ribonucleic acid (mRNA) since it carries the set of instructions contained in DNA. RNA is single stranded, the sugar in its nucleotide is ribose rather than deoxyribose as found in DNA, and the pyrimidine base U (uracil) is found in place of T (thymine). Also, U forms hydrogen bonds with A (adenine) in RNA. In transcription, a section of one strand of DNA corresponding to the gene is copied using the base complementarity, namely A-U and G-C. DNA microarray specifically aims to quantify the expression of genes by measuring their transcript levels. Typically a microarray is a glass or polymer slide, onto which DNA molecules are attached at fixed locations called spots. There are tens

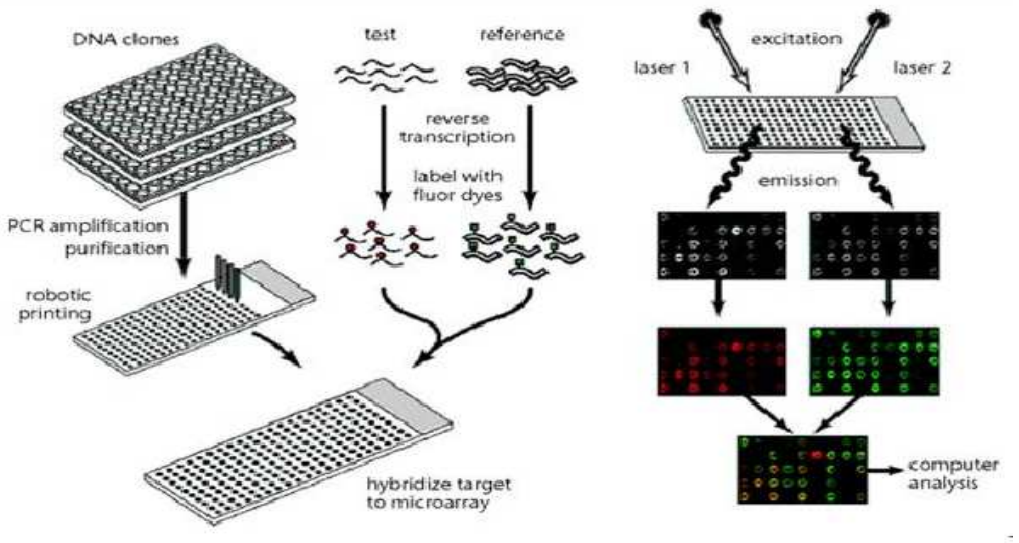

Figure 2: Microarray experiment procedure

of thousands of spots on one array, each containing tens of millions of identical DNA molecules. The experiment consists of measuring the expression of a gene in a cell by determining the amount of mRNA present. The novelty of the microarray is that it quantifies transcript levels on a global scale by quantifying transcript abundance of thousands of genes simultaneously.

The cDNA microarray experiment procedure is shown in Figure 2. We construct the microarray and obtain the DNA sequences representing genes of interest from two samples (the experimental and reference samples), transcribe the mRNA into more stable cDNA and add fluorescent labels. In practice, cDNA from the experimental and reference samples are labeled with different fluorescent dyes (usually green for reference sample and red for target sample), mixed and hybridized to probes on the array. The hybridized microarray is excited by a laser and scanned at wavelengths suitable for the detection of the red and green dyes. The amount of fluorescence emitted corresponds to the amount of nucleic acid bound to each spot. If the nu-

cleic acid from the experimental sample is in abundance, it will be red. The raw data that are produced from microarray experiments are digital images. To obtain information about gene expression levels, these images are analyzed, each spot on the array identified and its intensity measured and compared with values representing the background. These data which are extracted from the digitized image are combined into a spot quantification matrix. Each row corresponds to one spot on the array and each column represents different quantitative characteristics of that spot, such as the mean or median pixel intensity of spot and local background. An experiment typically consists of one or more spot quantification matrices representing all of the arrays. In the gene expression matrix, columns represent individual array samples and rows represent the genes and their measurements across all the arrays.

1.3.2 Protein

Proteins are built up by amino acids that are linked by peptide bonds to form a polypeptide chain. All of the 20 amino acids have in common a central atom (C_α) to which is attached a hydrogen atom (H), an amino group (NH_2) and a carboxyl group ($COOH$). The side chain attached to the C_α distinguishes one amino acid from another. In a polypeptide chain, the carboxyl group of an amino acid has formed a peptide bond, $C - N$, to the amino group of the next amino acid. One water molecule is eliminated in this process. The repeating units, which are called residues, are divided into main-chain atoms and side chains. The basic repeating unit along the main chain ($NH - C_\alpha H - C' = O$) is the common part among amino acids after peptide bonds have been formed. The main-chain (or backbone) part, which is identical in all residues, contains the central C_α atom attached to an NH group, a carboxyl group, and an H group. The side chain R, which is different for different residues, is bound to the C_α atom.

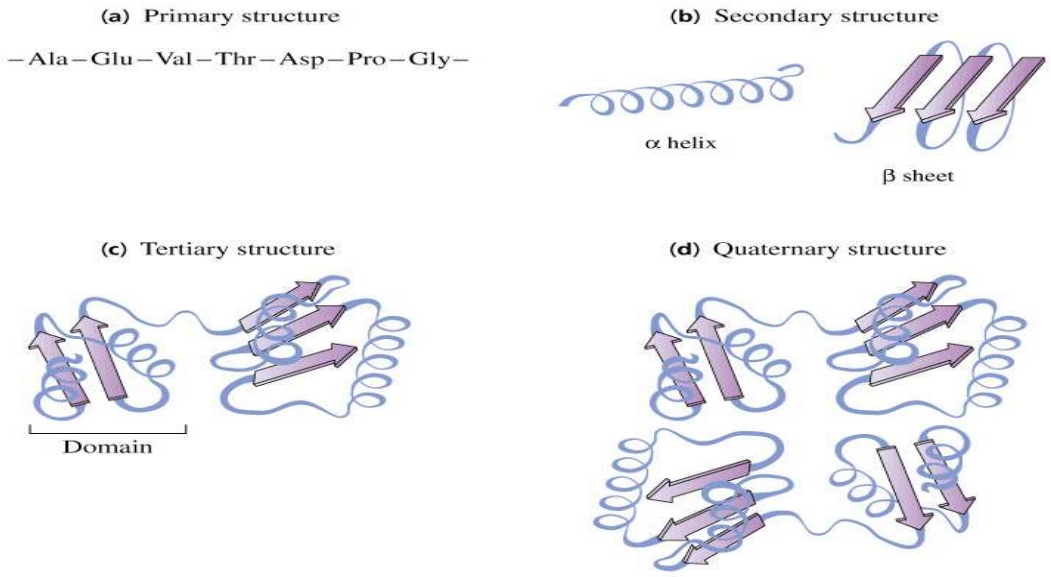

Figure 3: Protein structure

The amino acid sequence of a protein's polypeptide chain is called its primary structure. Different regions of the sequence form local regular secondary structures, such as alpha (α) helices or beta (β) strands. The tertiary structure is formed by packing such structural elements into one or several compact globular units called domains. The final protein may contain several polypeptide chains arranged in a quaternary structure. By formation of such tertiary and quaternary structures amino acids far apart in the sequence are brought close together in three dimensions to form a functional region, which is an active site. The different level of protein structure is illustrated in Figure 3. There are two main types of secondary structure, α helices and β sheets. Protein structures are built up by combinations of secondary structural elements. These form the core regions (the interior of the molecule) and they are connected by loop regions at the surface. α helices and β strands that are adjacent in the amino acid sequence are also usually adjacent in the three-dimensional structure. Simple combinations of a few secondary structure elements with a specific geometric

arrangement have been found to occur frequently in protein structures. These units have been called either supersecondary structures or motifs. Polypeptide chains are folded into one or several discrete units, domains, which are the fundamental functional and three-dimensional structural units. The cores of domains are built up from combination of small motifs of secondary structure, such as $\alpha - loop - \alpha$, $\beta - loop - \beta$, or $\beta - \alpha - \beta$ motifs.

1.4 Outline of Dissertation

The remainder of this dissertation is organized as follows. Chapter II describes the variable selection methods in cDNA microarray data using a multivariate Bayesian regression model. Chapter III reviews previous work in the field of the prediction of protein domain/linker regions and describes the data we construct from a protein database. Chapter III also introduces the use of a Hidden Markov model to model functionally or structurally different linker/non-linker regions in a protein sequence. In Chapter IV, we develop a model of these structurally different inter-domain linker/non-linker regions in a protein sequence using a non-stationary hidden Markov model. Chapter V provides some concluding remarks as well as a perspective on future work.

CHAPTER II

GENE SELECTION USING A TWO-LEVEL HIERARCHICAL BAYESIAN MODEL

2.1 Introduction

Microarray experiments typically measure the expression levels of several thousands of genes simultaneously. In cDNA data, it is common to have a large number of genes and a relatively small sample size. By removing redundant variables (genes), it would be possible to highlight those genes which are most relevant for certain events (say, certain diseases or a certain type of tumor).

Several approaches for finding the differentially expressed genes have been proposed: the T-test (e.g. Devore and Peck, 1997), a regression modeling approach (Thomas, Olson, Tapscott, and Zhao, 2001),a mixture model approach (Pan, 2002) and nonparametric methods (Troyanskaya, Garber, Brown, Botstein, and Altman, 2002). All of these are univariate gene selection methods, so suffer from the fact that no correlations between the genes are considered in the selection procedure. Recently Lee, Sha, Dougherty, Vanucci, and Mallick (2003) developed a multivariate Bayesian model to perform variable selection. Their method made use of a mixture prior distribution which is very sensitive toward the choice of some hyper-parameters, such as the mixing probability π. In general, the algorithm is slow due to complicated mixing structure of the posterior distribution.

From a machine learning viewpoint, high dimensionality and sparsity of data points suggest the use of Support Vector Machines (SVM) (Campbell, 2002). Usually SVMs achieve low test error despite small sample sizes. Several papers have reported results on the application of SVMs for performing variable selection (Guyon, Weston,

Barnhill, and Vapnik, 2002; Weston, Mukherjee, Chapelle, Pontil, Poggio, and Vapnik, 2001). However, this method has a number of disadvantages, such as the absence of probabilistic output and the necessity of estimating a trade-off parameter in order to utilize Mercer kernel functions. An alternative approach is to exploit the Bayesian technique of Automatic Relevance Determination (ARD). An ARD approach has been used previously for constructing a sparse classifier using the Relevance Vector Machine (RVM) of Tipping (2000). Li, Campbell, and Tipping (2002) utilized ARD to perform variable selection rather than using generalization bounds from statistical learning theory. Their variable selection method has similar performance to SVMs when applied to gene expression datasets from cDNA microarray data. The advantage of their approach is that variable sparsity is naturally incorporated into the algorithm - the optimal number of relevant variables is decided automatically. By contrast, for an SVM an additional variable selection procedure has to be added, and a further criterion must be used to indicate when the best variable set has been found. In terms of practical application, Li et al. (2002) highlight the importance of small number of influential genes. They use a zero-mean Gaussian prior with unknown variance for the unknown regression parameter β, which favors sparseness in estimating β. This choice of prior for β shows very good performances (Williams, 1998; Williams and Barber, 1998), but the main disadvantage is that it does not control the structural complexity of the resulting functions. That is, if one of the components of β happens to be irrelevant, a Gaussian prior will not set it exactly to zero, but instead to some small value (shrinkage rather than selection).

In this dissertation we consider a multivariate Bayesian regression model, and assign priors that favor sparseness in terms of number of variables (genes) used. We introduce the use of different priors to promote different degrees of sparseness using a unified two-level hierarchical Bayesian model. In our first model, we assign a zero

mean Gaussian prior to β with an independent prior distribution for the unknown variance of β. This model is related to ARD, though we perform full Bayesian analysis rather than marginal likelihood maximization. We use a Laplace prior in our second model, as it is known to promote sparseness (Williams, 1995), and is equivalent to the Lasso model. Our third model is based on the non-informative Jeffreys prior suggested by Figueiredo (2001). This particular prior does not contain any hyper-parameter, so that we can implement variable selection automatically, as well as strongly induce sparseness in the model. Importantly, the number of selected genes is decided automatically. Unlike other approaches, which are based on approximations, we will perform full Bayesian analysis exploiting simulation based on Markov Chain Monte Carlo (MCMC) methodology (Gelfand and Smith, 1990; Gilks et al., 1996) to derive the estimates (as well as the uncertainty distributions) of the unknown parameters.

We apply our methods to a leukemia data set from Golub, Slonim, Tamayo, Huard, Gaasenbeek, Mesirov, Coller, Loh, Downing, Caligiuri, Bloomfield, and Lender (1999) and also to a dataset from Hendenfalk, Duggan, Chen, Radmacher, Bittner, Simon, Meltzer, Gusterson, Esteller, and Raffeld (2001). The idea is to identify a small number of genes having the greatest discriminating power, thereby allowing researchers to quickly focus on the most promising candidates for diagnostics and therapeutics.

2.2 Model

Suppose that n independent binary random variables (e.g. normal and cancer), $Y_1, \cdots, Y_n$ are observed. $Y_i = 1$ indicates that sample i is cancer or one type of cancer (e.g. ALL, BRCA1) and $Y_i = 0$ indicates that sample i is normal or the other type of cancer (e.g. AML, BRCA2 and sporadic). For each sample we measure gene

expression levels for a set. Let X_{ij} denote the gene expression level of the jth gene for the ith sample and we form the data matrix $\mathbf{X}$ as

$$\mathbf{X} = \begin{pmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{np} \end{pmatrix}.$$

We define the binary regression model as $p_i = P(Y_i = 1) = \Phi(\mathbf{X_i}\beta), \quad i = 1, \cdots, n,$ where β is the vector of unknown regression parameters, $\mathbf{X_i}$ is the ith row vector of the matrix $\mathbf{X}$ and Φ is the standard normal cumulative density function linking the probability p_i with the linear structure $\mathbf{X_i}\beta$. This is known as the probit model.

Albert and Chib (1993a) introduce n independent latent variables $\mathbf{Z} = (Z_i, \cdots, Z_n)$ into the problem, where $Z_i \sim N(\mathbf{X_i}\beta, 1)$ and define $Y_i = 1$ if $Z_i > 0$ and $Y_i = 0$ if $Z_i \leq 0$. This approach connects the probit binary regression model for Y_i to a normal linear regression model for the latent variable Z_i.

We consider different priors for β in a two-level hierarchical Bayesian model. This model involves a zero mean Gaussian prior for β with unknown variances. We then assign choices of priors for the variances assuming they are independent. The prior distribution of β is

$$\beta|\mathbf{\Lambda} \sim N(\mathbf{0}, \mathbf{\Lambda})$$

where, $\mathbf{0} = (0, \cdots, 0)'$, $\mathbf{\Lambda} = diag(\lambda_1, \cdots, \lambda_p)$ and λ_i is the variance of β_i. We assign three different choices of prior distributions for Λ, which generates three different models inducing different degrees of sparsity to select the number of genes used.

2.2.1 Prior Distribution for the $\mathbf{\Lambda}$

For Model I, we assign a conjugate Inverse Gamma prior for each λ_i in $\mathbf{\Lambda}$ as $IG(\frac{a}{2}, \frac{2}{b})$. Here a random variable X is said to follow Inverse Gamma distribution if

$IG(\frac{a}{2}, \frac{2}{b}) \sim (\frac{1}{X})^{\frac{a}{2}+1} exp(-\frac{b}{2X})$. Note that we have two hyper-parameters, a and b, to be adjusted. We usually adjust a and b in such a way that the variance of λ is very large. This model is equivalent to ARD model of Li et al. (2002). Assuming independence among λ_is, the prior distribution of $\mathbf{\Lambda}$ is given by

$$\mathbf{\Lambda} \sim \Pi_{i=1}^{p} IG(\frac{a}{2}, \frac{2}{b}).$$

In Model II, we assign a Laplace prior for β to promote sparseness (so that irrelevant parameters are set exactly to zero). We can express the Laplace prior distribution as a scale mixture of Normal priors, which is equivalent to a two-level hierarchical Bayesian model. The Laplace prior can be expressed as a zero-mean Gaussian prior with an independent exponentially distributed variance: The proof of (2.1) is in appendix A.

$$\pi(\beta_i|\gamma) = \int_0^\infty \pi(\beta_i|\lambda_i)\pi(\lambda_i|\gamma)d\lambda_i \sim Laplace(0, \frac{1}{\sqrt{\gamma}}) \tag{2.1}$$

We assign an exponential distribution for the prior distribution of λ_i, which is equivalent to assigning a Laplace prior for β. Here a random variable X is said to follow exponential distribution with parameter γ, and is denoted as $expon(\gamma)$ and expressed as $expon(\gamma) = \frac{\gamma}{2} exp(-\frac{\gamma x}{2})$.

The prior distribution of $\mathbf{\Lambda}$ (again with the assumption of independence among λ_i) is given by

$$\mathbf{\Lambda} \sim \Pi_{i=1}^{p} expon(\gamma)$$

Here again we need to fix the hyper-parameter γ in such a way that the variance of λ is high. This is similar to the Lasso model but has added flexibility due to choices of multiple λs, rather than a single one as in the Lasso method.

In Model III, we attempt to avoid the problem of fixing the hyper-parameters by

letting the prior distribution of $\mathbf{\Lambda}$ be a non informative Jeffreys prior as

$$\mathbf{\Lambda} \sim |\mathbf{I}(\mathbf{\Lambda})|^{\frac{1}{2}} = \Pi_{i=1}^{p} \frac{1}{\lambda_i}$$

As already shown in Figueiredo (2001) and in our experimental results, this prior strongly induces sparseness and yields good performance.

2.3 Computation

The posterior distribution is not available in explicit form so we use the MCMC method (Gilks et al., 1996), specifically Gibbs sampling (Gelfand and Smith, 1990) to simulate the parameters from the posterior distributions. The details of derivations are provided in the appendix.

The full conditional distribution of $\mathbf{Z}$ has a truncated normal distribution. The random variables $Z_1, \cdots, Z_n$ are independent with

$$\begin{aligned} Z_i|\beta, Y_i = 1 &\propto N(\mathbf{X_i}\beta, 1) \text{ truncated at the left by } 0 \\ Z_i|\beta, Y_i = 0 &\propto N(\mathbf{X_i}\beta, 1) \text{ truncated at the right by } 0. \end{aligned}$$

We generate random numbers Z_i using the optimal exponential accept-reject algorithm (Robert, 1999).

In the two-level hierarchical Bayesian model with zero mean Gaussian priors and independently distributed variances for β, the full conditional distribution of β is as follows.

$$\pi(\beta|Z, Y, \mathbf{\Lambda}) \propto N(\Sigma \mathbf{X}' Z, \Sigma) \tag{2.2}$$

where, $\Sigma = (\mathbf{X}'\mathbf{X} + \mathbf{\Lambda}^{-1})^{-1}$. We have used the Woodbury-Sherman-Morrison matrix identity to reduce the dimension of the matrix, from p to n. This makes the computation much faster because we have cDNA data which has a high dimensionality

corresponding to the small sample size ($n << p$).

$$\Sigma = \mathbf{\Lambda} - \mathbf{\Lambda X}'(\mathbf{X\Lambda X}' + \mathbf{I})^{-1}\mathbf{X\Lambda}.$$

The full conditional distribution of $\mathbf{\Lambda}$ for the Inverse Gamma prior (Model I) is the following:

$$\pi(\mathbf{\Lambda}|Z, Y, \beta) \propto \Pi_{i=1}^{p} IG(\frac{a+1}{2}, \frac{2}{b+\beta_i^2}) \tag{2.3}$$

The full conditional distribution of $\mathbf{\Lambda}$ for the exponential prior (Model II) is the following:

$$\pi(\mathbf{\Lambda}^{-1}|Z, Y, \beta) \propto \Pi_{i=1}^{p} InvGauss(\frac{\sqrt{\gamma}}{\beta_i}, \gamma) \tag{2.4}$$

where $InvGauss$ denotes the inverse Gaussian distribution. The inverse Gaussian distribution for a random variable X is expressed as

$$InvGauss(\mu, \lambda) = \sqrt{\frac{\lambda}{2\pi x^3}} exp(-\frac{\lambda}{2\mu^2}\frac{(x-\mu)^2}{X}), \quad X \geq 0$$

We use the algorithm of Michael, Schucany, and Haas (1976) to generate the random number from the inverse Gaussian distribution.

The full conditional distribution of $\mathbf{\Lambda}$ with the Jeffreys prior (Model III) is the following :

$$\pi(\mathbf{\Lambda}^{-1}|\mathbf{Z}, Y, \beta) \propto \Pi_{i=1}^{p} G(\frac{1}{2}, \frac{2}{\beta_i^2}) \tag{2.5}$$

where G is the Gamma distribution. The gamma distribution for a random variable X is expressed as $G(a, b) \sim x^{a-1}e^{-x/b}$.

In practice, many of the λ_i approach zero, implying those genes can be pruned from the model. During MCMC iteration we delete genes using the criterion $\lambda_i < 10^{-12}$ as in Li et al. (2002). Also we re-introduce a gene which has been eliminated

if it has large enough variance (10 or more), but we find little change in performance on varying this re-introduction bound.

Finally, we obtain the predictive classification of a new observation Y_{new}, conditioning on the gene expression level X using the Monte-Carlo estimate :

$$\hat{P}(Y_{new} = 1|X) = \frac{1}{m}\sum_{t=1}^{m} p(Y_{new} = 1|X, \beta^t, Z^t, \Lambda^t) \tag{2.6}$$

where β^t, Z^t, Λ^t are the MCMC samples from the posterior distribution.

2.4 Application of Gene Selection

2.4.1 Leukemia Dataset

We apply our method to the Leukemia data set which has been extensively studied by Golub et al. (1999). The authors gathered bone marrow or peripheral blood samples from 72 patients with either acute myeloid leukemia (AML) or acute lymphoblastic leukemia (ALL). The data is split into a training set consisting of 38 samples of which 27 are ALL and 11 are AML, and a test set of 34 samples, 20 ALL and 14 AML. The gene expression levels for 7129 human genes are produced. Golub et al. (1999) investigated the use of a weighted voting scheme on the training samples and correctly classified 36 of the 38 training samples and also correctly classified 29 of the 34 test samples correctly, failing to predict correctly on 5. Using Golub's training data, we identify the 500 most significant genes by using two sample t-test statistics. We start with the 500 genes out of 7129, which include all the significant genes identified by Lee et al. (2003) and Li et al. (2002). We run the MCMC sampler (in our case, Gibbs sampling with 50,000 iterations and 20,000 burn-in). The priors are as follows. We assume $E(\lambda_i) = 10$ and $var(\lambda_i) = 100$ *a priori* for Model I and Model II and fix the hyper-parameters that way.

We obtain samples from the marginal posterior distribution and obtain the es-

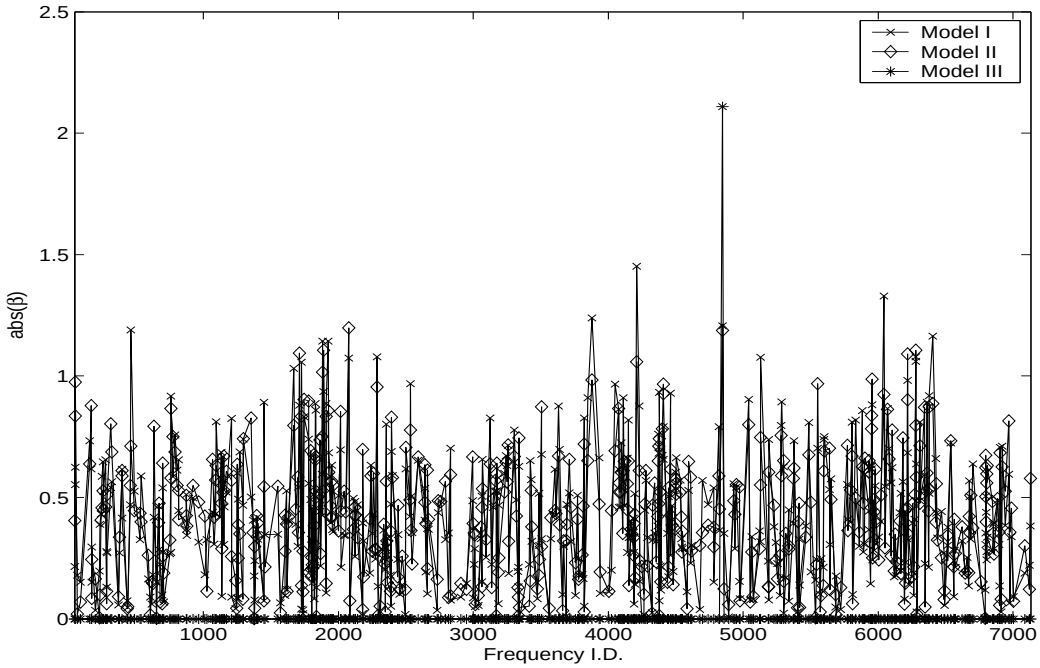

Figure 4: Leukemia data: Absolute value of β_i for three models

timates for β_is and λ_is. We plot the absolute values of β_i in Figure 4. Y-axis shows the absolute value of β_i and the x-axis shows the Frequency ID. The sparseness of β_i has been seen significantly in Model III. In addition, we can see that absolute values of β_i in Model I are usually bigger than those in Model II.

We select genes using the posterior variance of β. Variables with smaller variance will have no effect, and should be excluded from the model. Figure 5 shows the variance of β_i for each model. Y-axis shows the absolute value of β_i and the x-axis shows the Frequency ID. We can identify the genes having significantly larger variances than the others. Both Model I and II contain 20 genes which have significantly larger variance than the others. For Model I and Model II, we use these genes to perform prediction on the test data. The results are in Table 1 (we not only predict the correct classification but the probability related to it as well). There are 2 misclassifications (4th and 5th) by both Model I and Model II. The top 4 selected genes are com-

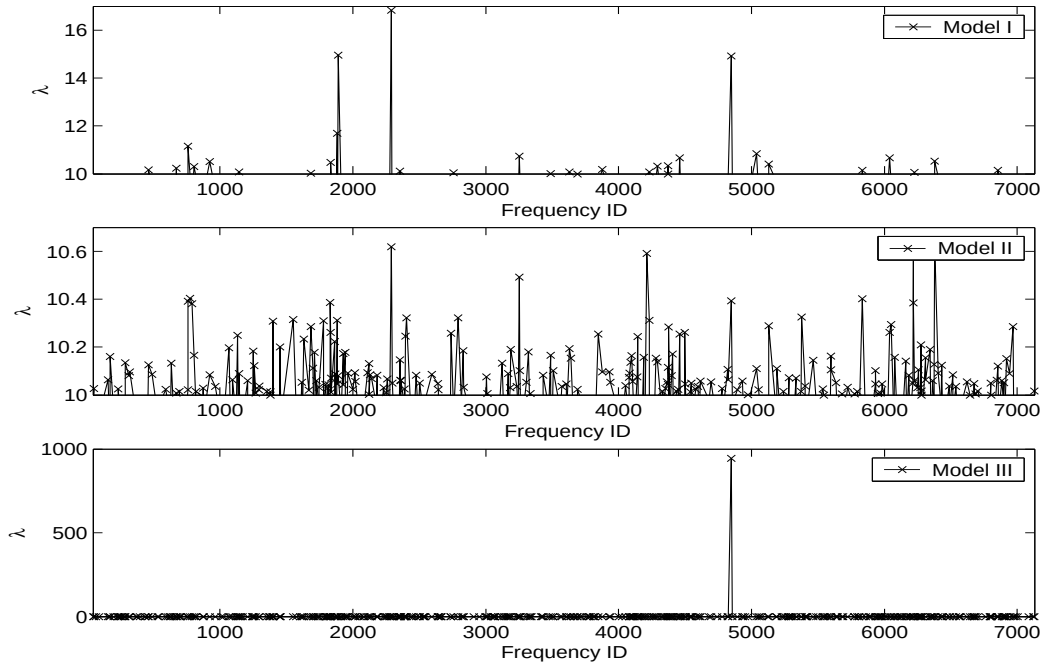

Figure 5: Leukemia data: The variance of β_i for three models

mon to both models: Zyxin (which encodes a LIM domain protein localized at focal contacts in adherent erythroleukemia cells, (Matsuda, Kawamura-Tsuzuku, Ohsugi, Yoshida, Emi, Nakamura, Onda, Yoshida, Nishiyama, and Yamamoto, 1996)); cell division control related protein (hCDCrel-1) mRNA (which is a partner gene of MLL in some Leukemias, (Osake, Rowley, and Zeleznik-Le, 1999)); HoxA9 mRNA (which collaborates with other genes to produce highly aggressive acute leukemia disease, (Thorsteinsdottir, Krosl, Hoang, and Sauvageau, 1999)) and MacMarcks (whose transcription is stimulated rapidly by tumor necrosis factor-alpha in human promyelocytic leukemia cells (Harlan, Graff, Stumpo, and Blackshear, 1991)).

In Model III, only Zyxin is selected, due to significantly larger variance than others. Zyxin has the third and second rank according to models I and II, respectively. The selected Zyxin is also one of leading genes in Lee et al. (2003) and Golub et al. (1999). Our prediction result based on Model III is in Table 1, which shows that

there are 3 misclassifications using only one gene. Golub et al. (1999) used 50 genes to predict, and had 5 misclassifications on test data. Our results appear to improve predictions done by Golub et al. (1999), having fewer misclassifications, while also using many fewer genes.

In these small data and high dimension problems, several models can fit the data well, each using a distinct set of genes. To investigate the issue, Li et al. (2002) randomly partition the data into two disjoint subsets of equal size and fit the model on both sets. After training they match the common number of genes to both models. This data is heterogeneous, as all 38 training samples were obtained from adult bone marrow, while some test samples came from peripheral blood or pediatric patients. This type of random partitioning and resampling of the data will make the data more homogeneous (Smith, Satagopan, Gonen, and Begg, 2002). Following this idea, we make new training and test data sets by randomly splitting the 72 samples in half (36+36 samples). We perform 50 re-samplings and select the top 20 genes. The top 20 selected genes for all the three models were in common at least 24% of times in the resampling results. The top four genes for model I and II were in common 50% to 70% of times. For model III, we found Zyxin was in common 80% of times.

This data is not very homogeneous, as observations were taken from different cells, so to control the variability we reanalyze on a subcategory of the data. For example, ALL cells can be either T-cells or B-cells. We apply our method to determine genes which are likely to be differentially expressed between ALL T-cells and ALL B-cells (Yeoh, Ross, Shurtleff, Williams, Patel, Mahfouz, Behm, Raimondi, Relling, Patel, Cheng, Campana, Wilkins, Zhou, Li, Liu, Pui, Evans, Naeve, Wong, and Downing, 2002). This way we control the heterogeneity of the sample type as much as possible by focusing on the B-cells and the T-cells experiments within the ALL group. This gives two reasonably homogeneous sample types, for which we still have

Table 1: Leukemia data: The prediction of the test data

Y	Model I $P(Y\|X_{\text{test}})$	Model II $P(Y\|X_{\text{test}})$	Model III $P(Y\|X_{\text{test}})$
1	1	1	1
1	1	1	1
1	1	1	1
1	0	0	1
1	0	0	1
1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
1	1	1	0
1	1	1	1
0	0	0	0
0	0	0	0
1	1	1	1
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
1	0.939	0.999	0
1	1	1	0
1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

many observations. We use 38 samples as training data set, and use the 9 samples as the test data set, which is the same procedure as Grant, Manduchi, and Stoeckert (2002). The results are in Table 2.

The top four selected genes in Model I are frequency ID 6855, 5542, 1882, 1962. The top selected gene, frequency ID 6855 is TCF-3 transcription factors (E2A immunoglobulin enhancer binding factors E12/E4). Heterodimers between TCF3 and tissue-specific basic helix-loop-helix (bHLH) proteins play major roles in determining tissue-specific cell fate during embryogenesis, like muscle or early B-cell differentiation (Kamps, Murre, Sun, and Baltimore, 1990). They are involved in a form of pre-B-cell acute lymphoblastic leukemia (B-ALL) through a chromosomal translocation which involves PBX1 and TCF3. T-cell Antigen CD7 precursor (frequency ID 5542) is one of two common selected genes by Dudoit, Yang, Callow, and Speed (2000) and Grant et al. (2002). The top four genes selected by Model II are frequency ID 6967, 1882, 6855, 4342. The top selected gene, frequency ID 6967 is SELL Leukocyte adhesion protein beta subunit (ITGB2). The ITGB2 protein product is the integrin beta chain beta 2. Integrins are integral cell-surface proteins composed of an alpha chain and a beta chain. A given chain may combine with multiple partners resulting in different integrins. For example, beta 2 combines with the alpha L chain to form the integrin LFA-1, and combines with the alpha M chain to form the integrin Mac-1. Integrins are known to participate in cell adhesion as well as cell-surface mediated signaling.

The gene TCF7 Transcription factor 7 (T-cell specific) (frequency ID 4342) is selected by the Model III. This gene is one of the two common selected genes by Dudoit et al. (2000) and Grant et al. (2002). The TCF7 gene encodes a transcription factor that is a member of the high mobility group protein family. Expression of TCF7 is specific to T cells, and the gene product was originally designated TCF-1, as a T-cell specific transcription factor . A closely related factor, LEF-1 (lympho-

cyte transcription factor), is expressed in both T and B cell lineages. Both TCF-1 and LEF-1 arise from the same gene, TCF7, by alternative splicing and use of dual promoters (Kingsmore, 1995).

Table 2: The selected genes for identifying T-cell vs. B-cell type

	Model I		Model I		Model I	
Rank	Variance	I.D.	Variance	I.D.	Variance	I.D.
1	11.588	6855	10.868	6967	13884.338	4342
2	10.874	5542	10.851	1882		
3	10.607	1882	10.798	6855		
4	10.595	1962	10.720	4342		
5	10.585	4017	10.698	760		
6	10.563	3233	10.685	5956		
7	10.562	760	10.631	5973		
8	10.560	4082	10.594	1095		
9	10.538	5973	10.592	2642		
10	10.501	6376	10.584	5976		
11	10.499	5171	10.582	2120		
12	10.483	5661	10.571	6376		
13	10.481	2642	10.571	2714		
14	10.473	1078	10.564	1685		
15	10.467	2714	10.552	4082		
16	10.464	4318	10.508	4547		
17	10.455	2121	10.499	1953		
18	10.444	2335	10.488	758		
19	10.426	412	10.480	407		
20	10.420	6127	10.464	4973		

2.4.2 Hereditary Breast Cancer Dataset

As a second study we also apply our method to a breast cancer dataset (Hendenfalk et al., 2001) from patients carrying mutations in the predisposing genes, BRCA1 or BRCA2, or from patients not expected to carry a hereditary predisposing mutation. Pathological and genetic differences appear to imply different but overlapping functions for BRCA1 and BRCA2. Hendenfalk et al. (2001) examined 22 breast tumor

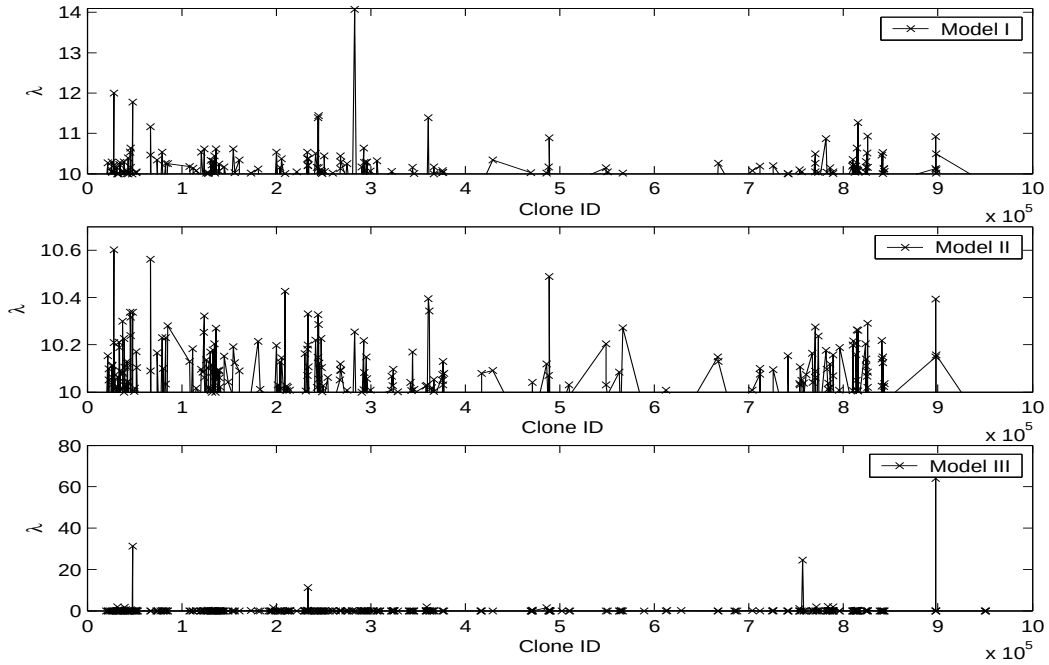

Figure 6: Breast cancer data: The variance of β_i for three models

samples from 21 breast cancer patients. Fifteen women had hereditary cancer, 7 having tumors with BRCA1 and 8 having tumors with BRCA2. 3226 genes were used for each breast tumor sample. We use our method to classify BRCA1 versus the others (BRCA2 and sporadic).

We use initial two-sample t-test statistics to identify the 500 most significant genes and run the MCMC sampler as in the previous example. We choose the same hyper-parameters as in the previous example. The variances of 500 genes are plotted it in Figure 6. Y-axis shows the absolute value of β_i and the x-axis shows the Frequency ID.

Some of the leading genes selected by these approaches appear among the 10 strongest genes in the list in Kim, Dougherty, Barrera, Chen, Bitter, and Trent (2002) and Lee et al. (2003). For both Model I and II, we select 25 genes which have significantly larger variances than others. The leading gene (by both of the approaches)

is keratin8 (KRT8), a member of the cytokeratin family of genes. Cytokeratins are frequently used to identify breast cancer metastases by immunohistochemistry, and cytokeratin8 abundance has been shown to correlate well with node-positive disease (Brotherick, Robson, Browell, Shenfine, White, Cunliffe, Shenton, Egan, Webb, Lunt, Young, and Higgs, 1998). Another top selected gene is tumor-associated antigen L-6 (TM4SF1), a member of a family of integral membrane proteins, several of which are also over expressed in tumors (Marken, Schieven, Hellstrm, Hellstrm, and Aruffo, 1992). Antigen L-6 is frequently over expressed in carcinomas, and antibody binding to L-6 on tumors in nude mouse models inhibits their outgrowth (Hellstrom, Beaumier, and Hellstrm, 1986).

In Model III only four genes appear to be the selected ones with significantly high variance. Keratin 8 and TM4SF1 are the top leading genes in Kim et al. (2002) and Lee et al. (2003) as well as in our previous two models. The other two genes are TOB1 and CTP syntheses and also appear in all the previously mentioned lists. The gene TOB1 interacts with the oncogene receptor ERBB2, and is found to be more highly expressed in BRCA2 and sporadic cancers, which are likewise more likely to harbor ERBB3 gene amplifications. TOB1 has an anti-proliferative activity that is apparently antagonized by ERBB2 (Matsuda et al., 1996).

We have checked the sensitivity (stability) of our analysis by adding a Gaussian noise to the expression values as in Lee et al. (2003). We re-analyzed the data contaminated by Gaussian noise to obtain the newly selected genes and have reproduced the results in Table 3. The results show that the analysis is quite stable, as it is selecting almost the same genes with a different noise level over the expression values. We also check the model adequacy by Leave-one-out Cross Validation (CV). We exclude a single data point, and predicted the $P(Y = 1|X)$ for that data point using equation (2.6). For Models I and II we use the 25 selected genes and for model III

the 3 selected genes to perform the cross validation. We compare the result of this cross validation with the observed response. There are 0 misclassification by model I and II and 2 misclassifications (17th and 18th sample) by Model III. We compare our cross validation results with other popular classification algorithms in Table 4. All other methods have used 51 genes. It is clear from the results that our methods improve the classification accuracy, having fewer misclassifications while also using fewer genes.

2.5 Discussion

We propose two-level hierarchical Bayesian models for variable selection which assume priors favoring sparseness in parameters. We employ latent variables to specialize the model to a regression model. We use simulation based MCMC methodology to derive the estimates of the unknown parameters. All three models provide good performance in terms of gene selection, but model III based on the Jeffreys prior is preferable, as there is no need to specify hyper-parameters or any type of threshold values. Simpler methods based on scores such as Fisher score or correlation coefficients can be used for gene selection but they usually select a much larger number of genes; the resulting small sample size may produce instability in the classification process. Due to the Bayesian setup, we have a coherent way to predict (assign) new samples to particular categories. Rather than hard rules (in or out) of assignment, we can evaluate the probability (chance) that the new sample will be in one of the categories, which is more helpful for decision making. Furthermore, the use of smaller number of important genes simplifies the experimental procedure.

Our gene selection method is based on the posterior mean of λ. We used informal, exploratory plots to find the genes with significantly large value of λ. A formal choice of cut off value to select significant λ based on posterior or predictive criteria will be

a topic of future research. All through our analysis, we assume data are independent and consider only binary classifiers. Future research will consider the gene with interaction situations and extend the analysis to multi-category models.

Table 3: Sensitivity analysis with breast cancer dataset

Rank	Model I	Model II	Model III
1	825478 *	360885 *	47884 *
2	28012 *	43021	244227
3	815503 *	897781 *	133534
4	897781 *	232826 *	950369
5	108422	82976	344352
6	488801 *	154323 *	897781 *
7	703846	840702	
8	813823	66774	
9	232826 *	74119	
10	244764 *	815530 *	
11	180803	244227 *	
12	66697 *	79353	
13	810408	200136 *	
14	815530 *	28469 *	
15	813651	488801 *	
16	244955	47884 *	
17	843249	23019	
18	295798 *	842894	
19	84955	566887 *	
20	809784 *	205490	
21	32750	115111	
22	360885 *	241348 *	
23	139217	282980 *	
24	22798	46019 *	
25	42313	725680	

The numbers are clone I.D.
$*$ means a genes were already selected
in the original analysis.

Table 4: Feature selection for the breast cancer data

	Model	Cross-Validation Error*
1	Feed-forward Neural Networks (3 hidden neurons, 1 hidden layer)	1.5 (Average error)
2	Gaussian Kernel	1
3	Epanechnikov Kernel	1
4	Moving Window Kernel	2
5	Probabilistic Neural Network (r=0.01)	3
6	kNN(k=1)	4
7	SVM Linear	4
8	Perceptron	5
9	SVM Nonlinear	6

$*$: Number of Misclassified Samples

51 Features used in the paper 'Gene-Expression Profiles in hereditary Breast Cancer', Vol 344, NJE

CHAPTER III

PREDICTION OF PROTEIN INTER-DOMAIN LINKER REGIONS BY A HIDDEN MARKOV MODEL

3.1 Introduction

A goal of protein structure prediction is to take an amino acid sequence (represented as a sequence of letters) and predict the 3-dimensional conformation (tertiary structure) adopted by the protein in its native (folded) state. The *domain* is the fundamental unit of the protein. A structural domain is defined as a unit that can independently fold into a stable tertiary structure. Many structural domains evolve as independent units that are found in different combinations. Thus, the domain has alternatively been defined as an evolutionary unit. The identification of domains within a protein sequence is valuable in numerous applications. Knowledge of structural domain boundaries can allow determination of separate domains using X-ray crystallography or Nuclear Magnetic Resonance (NMR), which is often more successful than trying to solve whole proteins. Working with separate domains may also be useful in functional assays. Computational methods for clustering proteins based on sequence similarity perform better when sequences are fragmented into single domain units.

Automated methods to predict domain boundaries can be divided into two broad categories, based on the definition used for domain. In the first category, domains are regarded as compact, semi-independent units with a hydrophobic core. Some of these methods use atomic coordinates from protein 3-dimensional structures determined by NMR or X-ray crystallography (e.g. Holm and Sander, 1994; Islam, Luo, and Sternberg, 1995; Siddiqui and Barton, 1995; Swindells, 1995; Wernisch, Hunting, and Wodak, 1999; Taylor, 1999; Xu, Xu, and Gabow, 2000; Alexandrov and Shindyalov,

2003). Other methods in this category rely on predicted secondary or tertiary structure instead of experimentally determined structure (George and Heringa, 2002c; Marsden, McGuffin, and Jones, 2002). A method that uses the structural domain definition, but does not depend on known or predicted structure, relies on residue side chain entropy (Galzitskaya and Melnik, 2003). In the second category, the evolutionary definition of domain is applied. Methods in this category use regions of conservation in multiple or pairwise sequence alignments to identify domain boundaries (e.g. Sonnhammer and Kahn, 1994; Gouzy, Eugene, Greene, Kahn, and Corpet, 1997; Gracy and Argos, 1998; Enright and Ouzonis, 2000; George and Heringa, 2002b; Liu and Rost, 2004a). The domain families in Pfam-A are created using profile hidden Markov models built on multiple sequence alignments (Bateman, Birney, Cerruti, Durbin, Etwiller, Eddy, Griffiths-Jones, Howe, Marshall, and Sonnhammer, 2002). The neural network method of Murvai, Vlahovicek, Szepesvari, and Pongor (2001) also depends on sequence homology. More recently, a third category of methods has emerged; these combine the structural and evolutionary definition of domain. For example, the CHOP method cuts proteins into domain-like fragments using domain boundary information from proteins with known structure and from Pfam-A (homology-based) domains (Liu and Rost, 2004a). CHOPNet is a neural network method that does not rely on known homology or known structure, but uses as input both evolutionary information and predicted structure (Liu and Rost, 2004b). The DGS (Domain Guess by Size) method uses neither structural nor evolutionary information, makes domain boundary estimates based on the statistical distribution of protein and domain lengths in a representative set (Wheelan, Marchler-Bauer, and Bryant, 2000).

An alternative to identifying domain boundaries is to identify inter-domain linker boundaries. The *linker* is defined as a region between adjacent domains. Studies have

shown that linkers can play an essential role in maintaining cooperative inter-domain interactions (Gokhale and Khosla, 2000). An understanding of linker characteristics will aid the design of linkers that allow gene fusion in protein engineering. The composition and length of linkers have been shown to affect protein stability, folding and domain-domain orientation (e.g. Robinson and Sauer, 1998). As an alternative to predicting domains, the prediction of linker regions is useful in splitting multidomain proteins into single domains. Splitting multidomain proteins without altering domain folding properties enables structural analysis for large proteins and allows biochemical analyses to identify functional domains on the sequences.

Many studies of linker regions in various protein families have reported that linker regions lack regular secondary structure (e.g. Argos, 1990), but a recent study has identified two main types of linker: helical and non-helical (George and Heringa, 2002a). Studies agree that certain amino acids, (e.g. Ala, Pro and charged residues) are more prevalent in the linker regions than in the domain regions (Robinson and Sauer, 1998; George and Heringa, 2002a; Tanaka, Kuroda, and Yokoyama, 2003). Most methods for identifying linkers use predicted secondary structure, amino acid propensity, or a combination of the two. The advantage of all the linker methods is they do not require known secondary structure or known homology. One approach is to identify regions that lack secondary structure using algorithms such as SEG (Wootton, 1994) or GlobPlot (Linding, Russell, Neduva, and Gibson, 2003). Miyazaki, Kuroda, and Yokoyama (2002) have applied a neural network to predict the linker boundaries based on amino acid propensity, and found that linkers possess characteristics different from loops. The method of Tanaka et al. (2003) combines secondary structure prediction to identify loop regions with amino acid frequency to distinguish linker and non-linker loops. The Udwary-Merski algorithm (Udwary, Merski, and Townsend, 2002) combines three properties of linkers: low sequence conservation

identified by multiple sequence alignment, low secondary structure conservation and low hydrophobicity. DomCut, which predicts linker regions based on sequence alone, relies solely on amino acid propensity (Suyama and Ohara, 2003). This method simply defines a linker region to be one that has lower linker index values than a specified threshold value. Similar to the DomCut method, we will apply a linker index. However, we will employ a hidden Markov model to predict not only the linker regions, but also the boundaries of these regions.

Hidden Markov models (HMMs) have been employed in diverse areas of computational biology. Lander and Green (1987) used HMMs in the construction of genetic linkage maps. Churchill (1989) employed HMMs to distinguish coding regions from non-coding regions in DNA. Later, simple HMMs were used in conjuction with the EM algorithm to model certain protein-binding sites in DNA (Cardon and Stormo, 1992). Haussler et al. (1993) applied HMMs to the problem of statistical modeling and multiple alignment of protein families. HMMs have been used widely in gene prediction (Kulp et al., 1996; Burge and Karlin, 1997; Henderson et al., 1997). Asai, Hayamizu, and Onizuka (1993) have applied HMMs to the problem of predicting the secondary structure of proteins, obtaining prediction rates that are competitive with previous methods in some cases. Churchill and Lazareva (1999) suggested a Bayesian approach to the problem of DNA sequence multiple alignment. Schmidler *et al.* (2000, 2001) worked on the prediction of the secondary structure of a protein by using a generalized HMM with a Bayesian estimation method. The observations in the HMMs for protein structure prediction are recognized as strings of amino acids (categorical variables), forming the primary sequence of a protein.

In this dissertation, sequences are assumed to have a structure composed of regions, such that the structure is homogeneous within a region but may differ between regions. We assume that protein sequence data is produced by a hidden Markov model

and that compositional variation is likely to reflect functional or structural differences between regions. We wish to develop hidden Markov models to model functionally or structurally different linker/non-linker sequence regions. Each region is classified into one of a finite number of states (e.g. linker region and non-linker region) and we want to estimate the states given the observed protein sequence. Importantly, we recognize the protein sequence data as continuous data instead of categorical data, which differs from the existing HMMs in computational protein sequence analysis. Therefore, it is also important to find values which identify differences between linker and non-linker regions in a protein sequence. We calculate the probability for each residue being in the linker region, so that researchers can have better understanding of the protein sequence structure. This is an advantage over other methods that rely on amino acid propensity. The existing methods (Suyama and Ohara, 2003; Miyazaki et al., 2002) do not give probabilistic output. In the model, the initial state sequence must begin with the non-linker region state in a protein sequence. This assumption is made to avoid the problem, associated with label-switching, that the likelihood function is the same for all permutations of the states and their parameters. If the prior is symmetric for all permutations of the parameters, the posterior is also symmetric, which creates difficulties in summarizing joint posterior distributions by marginal distributions and estimating unknowns by their posterior means.

Parameter estimation in HMMs usually relies on maximum likelihood or the Bayesian approach. In the Bayesian approach, we consider the HMM as a mixture model with missing data. We can associate observation y_i with missing data z_i which represents the state from which y_i is generated (Robert and Mengersen, 1999). The EM algorithm was originally tailored for missing data structures, but the dependency between the states creates problems for the EM algorithm for mixture estimation. While the simulation of the missing data is straightforward for an independent struc-

ture, it is quite difficult to simulate from the distribution of missing data conditional on the observed data in HMMs. The use of a recurrent forward-backward formula, which is widespread in the literature for estimating HMM parameters, is time consuming and numerically sensitive. Instead, We employ an efficient Bayesian estimation of the model through MCMC methods (Gilks et al., 1996), particularly Gibbs sampling (Gelfand and Smith, 1990), to implement inferences. Gibbs sampling effectively reduces the problem of sampling from a high-dimensional distribution to sampling from a series of low-dimensional distributions.

3.2 Data

3.2.1 Data Preparation

We downloaded protein sequence data from the Pfam database release 14 (Bateman et al., 2002) to construct a representative dataset of multidomain protein sequences. Pfam-A is a collection of domain families created using profile HMMs built on multiple alignments of homologous proteins. Release 14 of the Pfam database contains protein sequences from Swiss-Prot release 43.2 and SP-TrEMBL release 26.2 (Boeckmann, Bairoch, Apweiler, Blatter, Estreicher, Gasteiger, Martin, Michoud, O'Donovan, Phan, Pilbout, and Schneider, 2003). The Pfam database provides protein sequence coordinates for Pfam-A domains identified in these proteins. Protein sequences that were annotated as containing transmembrane regions in the Pfam database were removed from the dataset. We define a linker as a sequence segment of 4 to 20 residues that connects two adjacent regions identified by Pfam as domains. The reasoning behind this length range is that an inter-domain segment longer than 20 residues may contain a domain that has not yet been identified, instead of being one long linker region. We also define non-linker regions as sequence segments excluding linker regions. We denote a whole sequence as Full. We use only protein

sequences whose entire length can be classified as linker or domain by our criteria, except we allow up to 20 non-domain residues at the termini of the sequences. By this procedure, we obtained 11968 sequences with at least one linker region (14339 linker, 28726 corresponding domains and 824 unique domain regions).

We removed redundancy in this dataset as follows. First, we grouped the 11968 proteins into homeomorphic families (identical domain organization). We performed an all by all sequence comparison of the 11968 sequences using FASTA (Pearson and Lipman, 1988). We then applied single-linkage clustering using criteria of E-value $\leq 10^{-6}$ and at least 80% alignment coverage. Some of the resulting single-linkage clusters contained sequences with different domain organizations, due to the transitive nature of single-linkage clustering. Therefore, instead of selecting only one sequence from each cluster, we selected one sequence from each domain organization within each cluster. We also removed 7 protein sequences which were significantly longer than the rest (> 1000 residues). By this procedure, we obtained 802 sequences with at least one linker region. These 802 sequences contained 993 linkers and 1988 corresponding domain regions from 376 unique Pfam-A domain families. The average number of residues (the length) in the linker and the domain regions were 11.24 and 141.38, respectively.

The distributions of lengths of linker is displayed in Figure 7 and that of domain regions is in Figure 8. The frequency of amino acids in the different regions of the protein sequences are given in Table 5. There are some amino acids for which the distribution is significantly different between the linker region and other regions (domain, non-linker). The distribution of amino acids in the linker database of George and Heringa (2002a) shows similar patterns, even though their definition of linker region is different. The relative frequency of individual amino acids were compared between linker region and other regions by a $z-$test. Amino acids whose frequency is

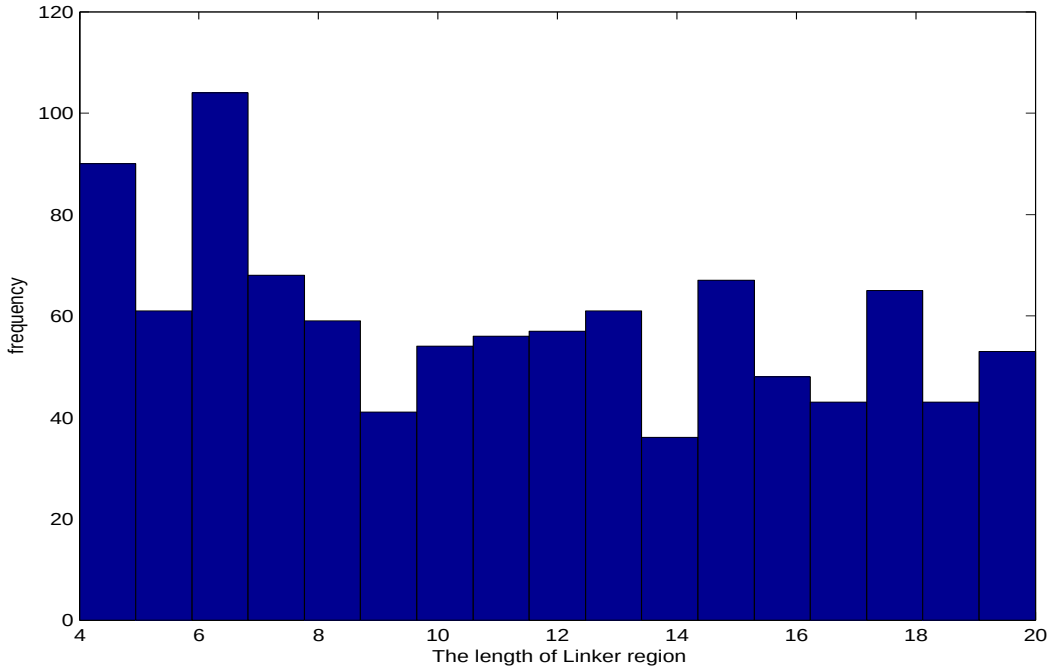

Figure 7: The distribution of the length of linker regions

significantly different between linker and other regions (P-value $< 10^{-3}$) are indicated by $(*)$ in Table 5. We incorporate this difference of amino acid composition between different regions within a protein by using the linker index.

3.2.2 Linker Index

Many studies have reported that certain amino acids may be observed with higher frequency in the linker regions than in the domain regions. Proline (P), Lysine (K), Glutamic acid (E), Serine (S), Aspartic acid (D) and Glutamine (Q) are preferred amino acids in linker regions. Previous studies (George and Heringa, 2002a; Suyama and Ohara, 2003; Tanaka et al., 2003) have shown Proline to be the most preferred linker amino acid. However, there is disagreement among these studies regarding which of the other preferred linker amino acids are significant. It is no surprise that Proline is favored because it has no amide hydrogen to donate in hydrogen bonding,

Table 5: The frequency of amino acids and the linker index

A. A.	Linker	Domain	Non Linker	Full	y_l^1	y_l^2	y_l^3
A	7.97 (7.94)	8.10	8.41	8.39	0.0166	0.0540	0.0515
C	0.89 (1.24)	1.50*	1.56 *	1.53 *	0.5724	0.5568	0.5370
D	6.32 (5.28)	5.60*	5.50 *	5.54 *	-0.1278	-0.1382	-0.1315
E	7.97 (6.89)	6.60 *	6.66 *	6.72 *	-0.1794	-0.1791	-0.1701
F	2.74 (4.34)	4.03 *	3.87 *	3.82 *	0.3561	0.3444	0.3309
G	7.74 (6.14)	7.37	7.35	7.37	-0.0442	-0.0516	-0.0491
H	1.91 (2.32)	2.27	2.24	2.22	0.1643	0.1560	0.1493
I	4.73 (5.13)	6.37 *	6.18 *	6.11 *	0.2758	0.2660	0.2552
K	6.97 (5.72)	5.81 *	5.65 *	5.71 *	-0.2134	-0.2104	-0.1997
L	7.51 (9.60)	9.54 *	9.60 *	9.51 *	0.2523	0.2460	0.2359
M	2.13 (2.15)	2.24	2.43	2.41	0.0197	0.1323	0.1266
N	4.22 (4.12)	4.08	3.91	3.93	-0.0786	-0.0755	-0.0719
P	6.63 (6.07)	4.30 *	4.46 *	4.56 *	-0.4188	-0.3964	-0.3742
Q	3.90 (4.05)	3.33 *	3.55	3.57	-0.1051	-0.0933	-0.0888
R	5.77 (5.79)	5.24	5.41	5.42	-0.0762	-0.0650	-0.0619
S	7.20 (5.55)	6.13 *	6.18 *	6.23 *	-0.1629	-0.1513	-0.1439
T	5.80 (5.66)	5.35	5.36	5.38	-0.0701	-0.0794	-0.0756
V	6.24 (6.64)	7.34 *	7.35 *	7.29 *	0.1782	0.1635	0.1565
W	0.81 (1.24)	1.32 *	1.18 *	1.16 *	0.3836	0.3703	0.3560
Y	2.46 (3.47)	3.38 *	3.06 *	3.03 *	0.2500	0.2188	0.2098

() : The frequency of a.a. in the Linker databases (George and Heringa, 2002a)

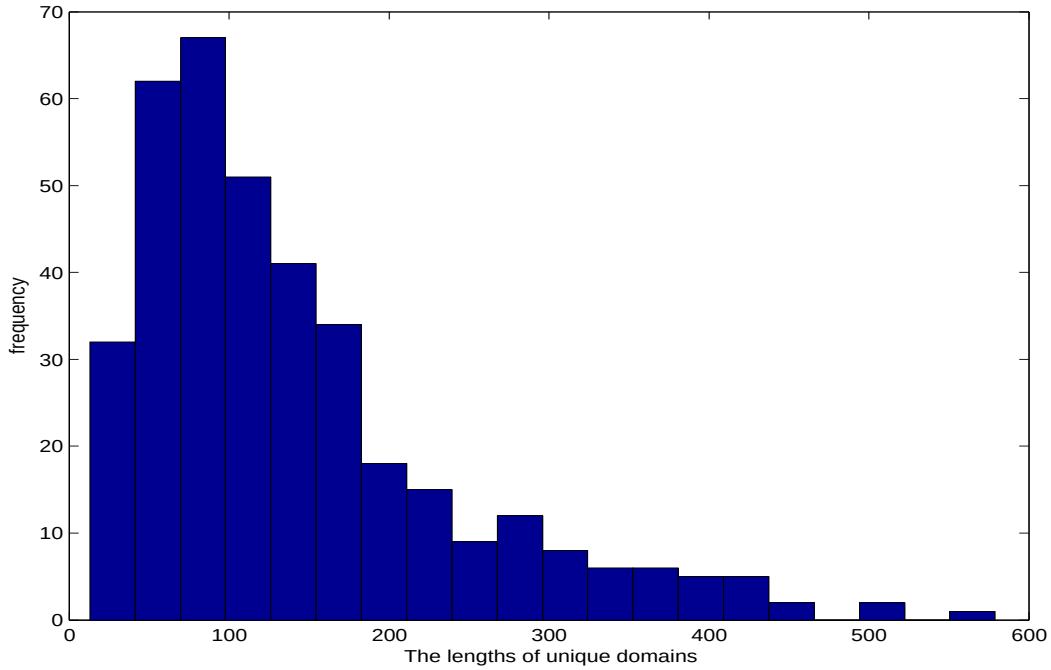

Figure 8: The distribution of the length of domain regions

and therefore structurally isolates the linker from domains (George and Heringa, 2002a). The analysis of our dataset also shows that Proline is the most preferred amino acid in the linker regions. We can find the propensity of amino acids between linker region and the other two regions. To incorporate the difference in amino acid composition between domain and linker regions, we employ the linker index, y_l^1, which reflects the preference of amino acids in the linker relative to the domain region, from Suyama and Ohara (2003).

$$y_l^1 = -ln(\frac{f_l^{linker}}{f_l^{domain}})$$

where, $f_l^{linker(domain)}$ is the relative frequency of the amino acid l in the linker (domain) region in the data set. Because y_l^1 represents the preference for amino acid l in the linker region, we note that the value of y_l^1 will be negative if the relative frequency of amino acid l in the linker region is greater than its relative frequency in the domain region. Similarly, we define linker index y_l^2 and y_l^3 which reflects the preference of

amino acids in the linkers relative to the non-linker region and the full sequence respectively.

$$y_l^2 = -ln(\frac{f_l^{linker}}{f_l^{non}}) \qquad y_l^3 = -ln(\frac{f_l^{linker}}{f_l^{full}})$$

where $f_l^{non(full)}$ is the relative frequency of the amino acid l in the non-linker/ full region sequences dataset. y_l^2 is similar with the statistics in Tanaka et al. (2003). The linker propensity of an amino acid in the linker region relative to the full sequence, y_l^3 , is defined by George and Heringa (2002a). To calculate the linker index for the sequences, we take an average of the linker index within each window size ω and assign this averaged linker index value, y_i to the center amino acid i of the window by sliding from the N-terminus to the C-terminus of a protein sequence. Since y_l^2 and y_l^3 give similar results and this is confirmed by computational results, we report only the results using y_l^1 and y_l^2. We use a window size, $\omega = 9$ which provides the greatest discrimination between the linker regions and the non-linker regions, and gives the best performance among the window sizes between 3 and 20. We constructed 2 datasets. One is the smoothed linker index data of protein sequences using y^1 (LD data) and the other is the smoothed linker index data of protein sequences using y^2 (LN data).

3.3 Model

Let $Y = (y_1, y_2, \cdots, y_n)'$ be the smoothed linker index data of a protein sequence generated by the corresponding hidden state $S = (s_1, s_2, \cdots, s_n)'$. Let the set of likelihood distribution parameters be θ and transition probability parameters be η. We assume two hidden states corresponding to the linker and the non-linker regions. If $s_i = 0$ then y_i is from a linker region and if $s_i = 1$ then y_i is from a non-linker

region.

$$s_i = \begin{cases} 1 & \text{if } y_i \in \text{Non-linker region} \\ 0 & \text{if } y_i \in \text{Linker region} \end{cases}$$

The transition probability matrix $\mathbf{P} = \{p_{lk}\}$ given by a two state HMM is

$$\begin{pmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{pmatrix} = \begin{pmatrix} p_{00} & 1 - p_{00} \\ 1 - p_{11} & p_{11} \end{pmatrix}$$

where $p_{lk} = p(s_i = k | s_{i-1} = l), \quad l, k \in \{0, 1\}$.

The problem is to infer the values of (S, θ, η) from the conditional joint posterior distributions $P(S, \theta, \eta | Y)$. The conditional joint posterior distributions is proportional to the joint distribution as follows

$$\begin{aligned} P(S, \theta, \eta | Y) &\propto P(Y, S, \theta, \eta) \\ &\propto P(Y, S | \theta, \eta) P(\theta, \eta) \\ &\propto P(Y | S, \theta, \eta) P(S, \theta, \eta) \\ &\propto P(Y | S, \theta, \eta) P(S | \theta, \eta) P(\theta, \eta) \end{aligned}$$

We assume the observed data y_i's are independent and have normal distribution. Both the mean and the variance of the observed data are parameterized in terms of the unobserved state variable with Markov process. We consider the following model for the i^{th} smoothed linker index in a protein sequence.

$$\begin{aligned} y_i &= \mu_0 + \mu_1 s_i + (1 + \omega s_i)^{1/2} \epsilon_i \qquad i = 1, \cdots, n \\ &\text{Restriction : } \quad \mu_1 > 0 \end{aligned}$$

where $\epsilon_i \sim N(0, \sigma^2)$ and ω denotes the proportional variance increase when $s_i = 1$. If y_i is from a linker region then it has the probability distribution as follows.

$$y_i | s_i = 0, \mu_0, \sigma^2 \sim N(\mu_0, \ \sigma^2)$$

If y_i is from a non-linker region then it has the probability distribution as follows.

$$y_i|s_i = 1, \mu_0, \mu_1, \omega, \sigma^2 \sim N(\mu_0 + \mu_1,\ \ (1+\omega)\sigma^2)$$

It is reasonable to give the restriction that the mean linker index of linker region (μ_0) is smaller than the mean linker index of non-linker region ($\mu_0 + \mu_1$) by the definition of linker index, because linker indices are negative for amino acids that are more prevalent in the linker region than other regions.

The likelihood distribution of Y given the corresponding state sequence S, $\theta = (\mu_0, \mu_1, \sigma^2, \omega)$ and $\eta = (p_{00}, p_{11})$ is

$$P(Y|S, \theta, \eta) \ \ = \ \ N((\underline{1}, S)\underline{\mu},\ \ \sigma^2\Sigma|S, \theta, \eta)$$

where, $\underline{\mu} = (\mu_0, \mu_1)'$, $\underline{1} = (1, \cdots, 1)$ is a $n \times 1$ vector and $\Sigma = diag((1+\omega s_1), \cdots, (1+\omega s_n))$.

We can assume $P(s_1 = 1) = 1$ because the initial state must begin with the non-linker region state in a protein sequence. Given the hidden state S, the transitions n_{ij} from state i to j are sufficient statistics for p_{00} and p_{11}. The likelihood distribution conditioned on the initial state being non-linker, $p(S_i = 1) = 1$, is given by

$$\begin{aligned} P(S|\theta, \eta) &= P(s_1|\theta, \eta) \prod_{i=2}^{n} P(s_i|s_{i-1}, \theta, \eta) \\ &\propto p_{00}^{n_{00}}(1-p_{00})^{n_{01}} \times p_{11}^{n_{11}}(1-p_{11})^{n_{10}} \end{aligned}$$

where n_{ij} is the number of observations of transitions from state i to j.

We need to specify the prior distribution $P(\theta, \eta)$ to complete the joint distribution $P(Y, S, \theta, \eta)$.

The primary interest $P(S|Y)$ can be obtained by integrating out the conditional

joint posterior distributions of $P(S, \theta, \eta|Y)$ with respect to θ.

$$\begin{aligned} P(S|Y) &= \int_\theta \int_\eta P(S, \theta, \eta|Y)\, d\theta\, d\eta \\ &\propto \int_\theta \int_\eta P(Y|S, \theta, \eta) P(S, \theta, \eta)\, d\theta\, d\eta \\ &\propto \int_\theta \int_\eta P(Y|S, \theta, \eta) P(S|\theta, \eta) P(\theta, \eta)\, d\theta\, d\eta \end{aligned} \tag{3.1}$$

Using equation (3.1), we can calculate the probability of being in a state k for each amino acid i in a protein sequence given $y_i, s_{i-1} = l, \theta$ and η.

$$\begin{aligned} P(s_i = k|y_i, s_{i-1} = l, \theta, \eta) &= \frac{P(y_i|s_i = k, s_{i-1} = l, \theta, \eta) P(s_i = k|s_{i-1} = l, \theta, \eta)}{P(y_i|s_{i-1} = l, \theta, \eta)} \\ &= \frac{P(y_i|s_i = k, \theta, \eta) p_{lk}}{P(y_i|\theta, \eta)} \quad \because\ y_i \text{ doesn't depend on } s_{i-1} \\ &= \frac{P(y_i|s_i = k, \theta, \eta) p_{lk}}{\sum_{j=0}^{1} P(y_i, s_i = j|\theta, \eta)} \\ &= \frac{P(y_i|s_i = k, \theta, \eta) p_{lk}}{\sum_{j=0}^{1} P(y_i s_i = j|\theta, \eta) p_{lj}} \end{aligned} \tag{3.2}$$

Once, these simulated sample values have been obtained from equation (3.2), any posterior moment or marginal distribution can be easily estimated. Specifically the posterior expectation can be estimated by the sample average, using equation (3.3).

$$E[P(s_i = k|y_i, s_{i-1} = l, \theta, \eta)] = \frac{1}{m} \sum_{t=1}^{m} \frac{P(y_i|s_i^{(t)} = k, \theta^{(t)}, \eta^{(t)}) p_{lk}^{(t)}}{\sum_{j=0}^{1} P(y_i|s_i^{(t)} = j, \theta^{(t)}, \eta^{(t)}) p_{lj}^{(t)}} \tag{3.3}$$

where, t denotes the iteration in MCMC sampler, $k \in \{0, 1\}$ and m is the number of the MCMC samples from the posterior distribution after burn-in. We predict the state of an amino acid using the classification variable CV_i.

$$CV_i = \begin{cases} 1 & \text{if } E[P(s_i = k|y_i, s_{i-1} = l, \theta, \eta)] \leq 0.5 \\ 0 & \text{if } E[P(s_i = k|y_i, s_{i-1} = l, \theta, \eta)] > 0.5 \end{cases} \tag{3.4}$$

3.3.1 The Prior Distributions

The Bayesian approach to inference requires specification of a prior distribution for the parameters of the model. We assign mutually independent prior distributions for $\underline{\mu}$ and σ^2. The prior of $\underline{\mu}$ is assigned to be the conjugated normal distribution and the prior of σ^2 is the inverse gamma distribution. Here a random variable X is said to follow Inverse Gamma distribution if $IG(\frac{a}{2}, \frac{2}{b}) \sim (\frac{1}{X})^{\frac{a}{2}+1} exp(-\frac{b}{2X})$.

$$\begin{aligned} \underline{\mu} &\propto N\left(\begin{pmatrix} \mu_{0a} \\ \mu_{1a} \end{pmatrix}, \begin{pmatrix} \xi_{0a}, & 0 \\ 0 & \xi_{1a} \end{pmatrix}\right) \\ \sigma^2 &\propto IG(\frac{a}{2}, \frac{2}{b}) \end{aligned}$$

where, $(\mu_{0a}, \mu_{1a}, a, b)$ are hyper-parameters to be adjusted. Given hidden state s_i, ω only depends on the observations for $s_i = 1$. We use the expression $\bar{\omega} = (\omega + 1)$ as in Albert and Chib (1993b) to make ω represent the proportionate increase in variance when $s_i = 1$. Let the prior distribution of $\bar{\omega}$ be the truncated inverse gamma distribution

$$\bar{\omega} \propto IG(\frac{a_w}{2}, \frac{2}{b_w}) \times I(\bar{\omega} > 1)$$

For the priors for (p_{00}, p_{11}), we assign the conjugate beta distribution.

$$P(p_{00}, p_{11}) \propto beta(u_{00}, u_{01}) \times beta(u_{11}, u_{10})$$

Here a random variable X is said to follow beta distribution if $beta(a, b) \sim X^{a-1}(1 - X)^{b-1}$.

3.4 Computation

The conditional joint posterior distribution is not available in explicit form, so we use the MCMC method, specifically Gibbs sampling to simulate the unknown parameters from the conditional joint posterior distribution.

It is convenient to transform the data to q_i by multiplying each observation y_i by $(1+\omega s_i)^{-1/2}$ so that the transformed data have constant variances, instead of variances that depend on the hidden state.

$$q_i y_i \sim N((q_i, q_i s_i)\underline{\mu},\ \sigma^2) \equiv y_i^* \sim N(w_i^* \underline{\mu},\ \sigma^2)$$

Define $Y^* = (y_1^*, \cdots, y_n^*)'$ and $W^* = (w_1^*, \cdots, w_n^*)'$. The derivations of the full conditional distributions are provided in the appendix. Here $\theta|\cdot$ denotes θ conditioning on all other parameters.

The full conditional distributions of $\underline{\mu}$ and σ^2 are following :

$$\underline{\mu}|\cdot \ \sim\ N\left(\mathbf{A}^{-1}(\sigma^{-2}W^{*'}Y^* + \mathbf{V}^{-1}\underline{\mu}_0),\quad \mathbf{A}^{-1}\right) \times I(\mu_1 > 0) \tag{3.5}$$

$$\sigma^2|\cdot \ \sim\ IG\left(\frac{a+n}{2},\quad \frac{2}{b + (Y^* - W^*\underline{\mu})'(Y^* - W^*\underline{\mu})}\right) \tag{3.6}$$

where, $A = (\mathbf{V}^{-1} + \sigma^{-2}W^{*'}W^*)$, $\underline{\mu}_0 = (\mu_{0a}, \mu_{1a})'$ and $\mathbf{V} = diag(\xi_{0a}, \xi_{1a})$.

The full conditional distribution of $\bar{\omega}$ is the truncated inverse gamma distribution.

$$\bar{\omega}|\cdot \sim IG\left(\frac{n_1 + a_w}{2},\ \frac{2}{\sum_{i\in J}\left[\frac{(y_i - \mu_0 - \mu_1 s_i)^2}{\sigma^2}\right] + b_w}\right) \times I(\bar{\omega} > 1) \tag{3.7}$$

where, $J = \{i | s_i = 1, i = 1, \cdots, n\}$ and n_1 is the number of observations whose state is 1.

It is obvious that we only need to consider the conditional distribution $P(p_{00}, p_{11}|S)$ since (p_{00}, p_{11}) is independent of (Y, θ) given S. The full conditional distributions of $\eta = (p_{00}, p_{11})$ are the following :

$$p_{00}|s \ \sim\ beta(n_{00} + u_{00} - 1, n_{01} + u_{01} - 1) \tag{3.8}$$

$$p_{11}|s \ \sim\ beta(n_{11} + u_{11} - 1, n_{10} + u_{10} - 1) \tag{3.9}$$

The full conditional distribution of $\{s_i, i = 1, \cdots, n\}$ depends on the state at

time $(i-1)$ and $(i+1)$ since s_i has a Markov property.

$$\begin{aligned}
P(s_i|\cdot) &\propto \frac{P(s_i|Y_i, S_{-i}) f(y_{i+1}, \cdots, y_n|Y_i, S_{-i}, s_i)}{f(y_{i+1}, \cdots, y_n|Y_i, S_{-i})} \\
&\propto P(s_i|Y_i, S^{-i}) \quad \because \text{y's are independent} \\
&\propto Pr(s_i|Y_{i-1}, S_{i-1}) f(y_i, s_{i+1}, \cdots, s_n|Y_{i-1}, S_i) \\
&\propto P(s_i|s_{i-1}) f(y_i|Y_{i-1}, S_i) P(s_{i+1}|Y_i, S_i) P(s_{i+2}, \cdots, s_n|Y_i, S_{i+1}) \\
&\propto P(s_i|s_{i-1}) f(y_i|Y_{i-1}, S_i) P(s_{i+1}|s_i) \\
&\propto P(s_i|s_{i-1}) P(s_{i+1}|s_i) f(y_i|s_i, \theta, \eta) \quad \text{for} \quad 2 \leq i \leq n-1
\end{aligned}$$

where, $S^{-i} = (s_1, \cdots, s_{i-1}, s_{i+1}, \cdots, s_n)'$, $Y_i = (y_1, \cdots, y_i)$ and $S_i = (s_1, \cdots, s_i)$. Note that $p(s_1 = 1) = 1$ and $p(s_n = 1) = 1$ because a protein sequence always begins and ends with a linker region. The derivations of full conditional distributions are in appendix.

3.5 Results

We applied our model to two protein sequence datasets which we constructed from Pfam-A release 14 using linker index y^1 (LD dataset) and linker index y^2 (LN dataset) as described in section 3.2. We divided each dataset into a training dataset and a test dataset randomly with the ratio of 4:1. We trained the model with the training dataset of 642 sequences and tested the trained model with the test dataset of 160 sequences. We ran an MCMC sampler, particularly Gibbs sampling with 40,000 iterations and 10,000 burn-in. The choice of hyper-parameters, which are based on the data and the problem at hand, are as follows. We let hyper-parameters for $\underline{\mu}$ be the sample means of the training dataset for each state and give each a sufficiently large variance of 10. We assume $E(\bar{\omega}) = 1.5$, $\text{var}(\bar{\omega}) = 10$ and $E(\sigma^2) = 0.1$, $\text{var}(\sigma^2) = 10$ and fix the hyper-parameters accordingly. We let the hyper-parameters for the p_{00} and p_{11} be

$u_{ij} = 1$, $i, j \in \{0, 1\}$ to give uniform priors. The posterior distributions of estimates are provided in appendix B.

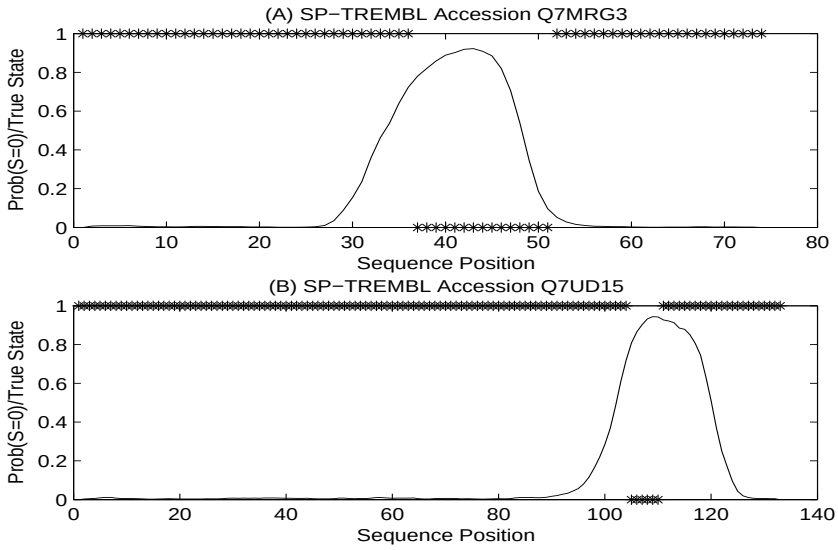

Figure 9: Examples of good predictions using LD dataset

We calculated the probability of being in the linker region, $p(s_i = 0|y_i)$, for each residue i along a protein sequence. Figure 9 shows a case with good prediction, in which probabilities in the linker region are much higher than in other regions. In Figure 9, $* = 1$ denotes those residues are in the non-linker region and $* = 0$ denotes those residues are in the linker region. However, we need to select a cut-off value (here, 0.5) to delineate the boundary.

Although our method gives high probabilities to the linker region, it also gives high probabilities to some regions that are not linker regions, but may have similar structure. Most linker regions have negative linker index values, but some non-linker regions also have this pattern. Figure 10(A) shows one of these cases. In Figure 10, $* = 1$ denotes those residues are in the non-linker region and $* = 0$ denotes those residues are in the linker region. There are two regions with high probability, but there

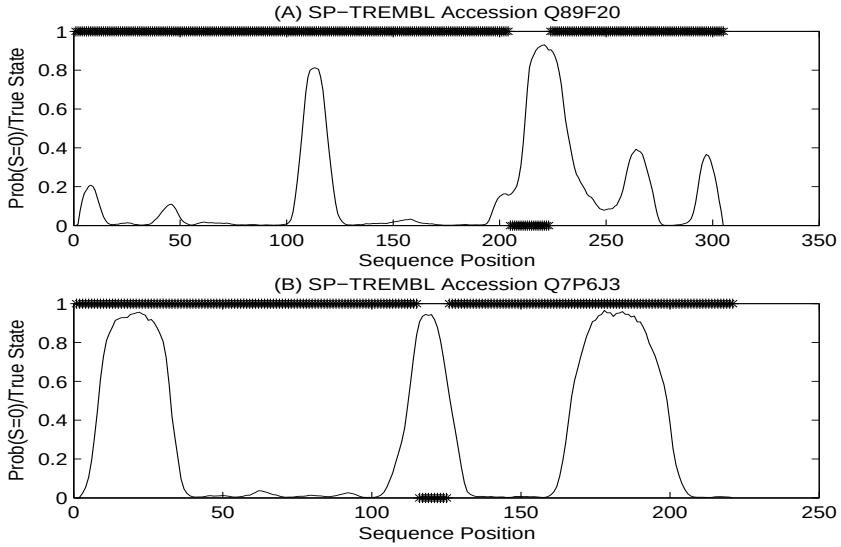

Figure 10: Examples of overpredictions using LD dataset

is only one linker region in the protein. However, the probability of the actual linker region is higher than that of the other putative linker region. Figure 10(B) shows that the termini of the sequence can have high probabilities. The characteristics of the termini are not known. This case indicates that termini have patterns similar to those in the linker region.

In this case, we need not take the high probabilities at the termini of the protein sequences naively; since we know that a linker region is an inter-domain region, we can avoid incorrectly predicting termini to be linkers when using our method. When applying our method to the test data of the LD and LN datasets to predict the linker region, we used a whole sequence in the prediction instead of part of it. Other studies (Miyazaki et al., 2002; Suyama and Ohara, 2003) ignore the termini of a sequence or unknown regions in a sequence in the prediction.

We classify *every* residue into one of two states (linker or non-linker) using the classification variable (CV) which is based on the probability in (3.3).

Table 6: The evaluation of the model

Evaluation	LD data	LN data
Sensitivity(Sn)	80.68	81.31
Specificity(Sp)	56.27	55.62
Correlation(C)	65.23	65.08

To evaluate our method for the prediction of linker regions, sensitivity (Sn), specificity (Sp) and correlation coefficient (C) are reported with the test datasets. TP refers to those amino acids that are correctly labeled as linker and FP refers to amino acids that are labeled as linker while in fact they are non-linker. FN refers to amino acids that are labeled as non-linker while in fact they are linker. TN refers to amino acids that are correctly labeled as non-linker. The sensitivity is the percentage of actual linker residues that were predicted to be linker ($Sn = \frac{TP}{TP+FN}$) and the specificity is the percentage of predicted linker residues which are truly linker ($Sp = \frac{TP}{TP+FP}$). The correlation coefficient (Matthews, 1975) is an indication of how much better a given prediction is than a random one $\left(C = \frac{(TP)(TN)-(FN)(FP)}{\sqrt{(TP+FN)(TP+FP)(TN+FN)(TN+FP)}}\right)$. $C = 1$ indicates perfect prediction, while $C = 0$ is expected for a prediction no better than random. The evaluation results are reported in Table 6. The evaluation results for LD and LN data appear to be similar. Therefore, there is little difference between using linker index value y^1 and y^2 in the ability to distinguish linker and non-linker regions in a protein sequence. Sensitivities of the model using both LD and LN datasets are about 81%, which indicates that we can predict 81% of the residues in the linker regions correctly out of all the residues in the linker regions using our method. However, specificities are about only about 56%, which indicates out of all of the residues predicted as being in the linker region only 56% are in cat really in the linker region (there are many false positives). As we have seen from Figure 8, there are regions such as termini that have similar structure compared with linker regions.

Those regions cause overpredictions of linker regions, resulting in false positives. The Matthew's correlation coefficients are about 65%.

Table 7: The evaluation and the comparison with DomCut

	LD dataset	LN dataset	DomCut
Sensitivity	63.29	63.77	56.52
Selectivity	92.91	88.59	87.97
Correct Linker	131(141)	132(149)	117(133)

*: There are 207 linker regions in the test data.
*: DomCut use LD dataset as linker index.
*: () is the number of predicted linker regions.

There are several other methods for predicting protein linker regions. It is difficult to directly compare the accuracy of the predictions between methods, because each of these methods uses a different definition of linker region and criteria for assessing the method. We compare our method with DomCut (Suyama and Ohara, 2003) because the authors use a similar definition of linker region and also use linker index, which is the same as our y^1, to reflect the difference in amino acid composition between domain and linker regions. The authors of DomCut estimate sensitivity as the proportion of the total number of correctly predicted linker regions against the total number of linker regions and they estimate selectivity as the proportion of correctly predicted regions in all predicted regions (Suyama and Ohara, 2003). This is different than our measures of sensitivity and selectivity described above, which consider predicted linker residues, rather than predicted regions. DomCut predicts putative linker regions instead of giving specific boundaries of linker region, and is a good intuitive method rather than a statistical method. We applied the DomCut method to our LD test datasets with various criteria and report the best result in the Table 7. In order to compare our method with the DomCut method, we also excluded the unknown regions in the sequences in the evaluation, as was done in the evaluation

of DomCut (Suyama and Ohara, 2003). In DomCut, a linker region is taken to be correctly predicted if there is a trough in the linker region and the minimum linker index value is lower than the cut-off. The smoothed linker index was calculated with window size 9 and the cut-off is -0.08. In order to compare, we used their definition of the sensitivity and the selectivity, based on prediction of linker regions, rather than residues. In our method, a linker region is taken to be correctly predicted if there is a region that has a consecutive high probability (> 0.5), its length is longer than 4 residues, and the maximum probability is higher than 0.8. The sensitivity and the selectivity of our method using LD data are 63.3% and 92.9% respectively and these are slightly better than those of our method using LN data. This result appears to improve the DomCut method, even though it is difficult to directly compare.

3.6 Discussion

We have developed a hidden Markov model to model inter-domain linker/non-linker regions in a protein sequence using the composition differences of amino acids between the two different regions. We take sequence data as continuous data, instead of categorical data, using linker index value to distinguish linker and non-linker regions. We also calculate the probability of being in the linker region at each residue along a protein sequence. An advantage our HMM approach has over existing methods that do not give probabilistic output is that we may use the results to gain further insight into properties of the primary sequence. Furthermore, the probabilities can be considered when attempting to fragment unknown proteins into domains using molecular techniques prior to structure determination by NMR or X-ray crystallography. An advantage of both our method and DomCut is that they can be applied to both the structural and evolutionary definitions of domain, depending on the dataset used, because neither relies on structure prediction.

Not only is the composition of the linker region important, but also its length. In general, altering the length of linker regions connecting domains has been shown to affect protein stability, folding rates and domain-domain orientation (VanLeeuwen, Strating, Rensen, Laat, and der Vliet, 1997; Robinson and Sauer, 1998). We may incorporate the duration probability density into HMMs to utilize the distribution of the lengths of the linker/non-linker region. Therefore, we may consider a HMM with a duration probability density (a variable duration HMM) to utilize the distribution of the lengths of the linker/non linker region. Also the state duration can be modeled by allowing all the transition probabilities to be functions of d, which is the duration of the same state stays.

CHAPTER IV

PREDICTION OF PROTEIN INTER-DOMAIN LINKER REGIONS BY A NON-STATIONARY HIDDEN MARKOV MODEL

4.1 Introduction

We have developed a model to predict linker/non-linker regions in a protein sequence by exploiting differences in the composition of amino acids between the two regions and using a conventional hidden Markov model.

A well known weakness of conventional HMMs is weak state duration modeling (see Figure 11). The inherent duration distribution in a conventional HMM is a geometric distribution. In many applications, a geometric duration model might not be appropriate, and it may be desirable to model the state duration with other distributions. The importance of incorporating state duration is reflected in the observation that, for some problems, the quality of the model is significantly improved when an explicit state duration distribution is used (Rabiner, 1989).

We can expect that the specified state duration modeling would improve protein linker prediction, because both the composition of the linker region and its length are important. In general, altering the length of linker regions connecting domains has been shown to affect protein stability, folding rates and domain-domain orientation (VanLeeuwen et al., 1997; Robinson and Sauer, 1998). We incorporate a state duration probability distribution into an HMM to utilize information about the distribution of the lengths of the linker/non-linker region. The distribution of the lengths of linker and non-linker regions does not follow a geometric distribution (see Figure 7 and Figure 8).

To overcome this limitation of conventional HMM, variable duration hidden

Markov models (VDHMM) were introduced by Ferguson (1980) (see Figure 12). The VDHMMs consider a Markov chain which produces several observations at a given state. Sometimes this is called a generalized HMM or a segmental HMM. Schmidler *et al.* (2000, 2001) worked on the prediction of the secondary structure of a protein by a generalized HMM with a Baysian estimation method. Most previous work on estimating the unknown states of the VDHMM in computational biology assumes the state transition probabilities are constant. A more complex variable duration model, where the state transition probabilities are modeled by functions of time duration, was introduced by Sin and Kim (1995). This model is referred to as a Non-stationary hidden Markov model (NSHMM)(see Figure 13).

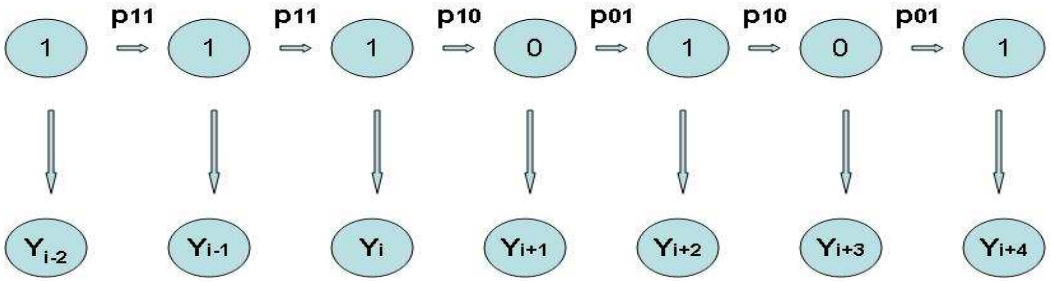

Figure 11: Representation of HMM

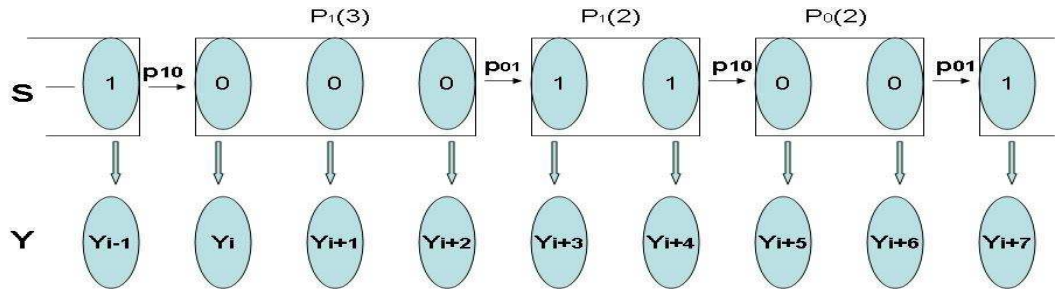

Figure 12: Representation of VDHMM

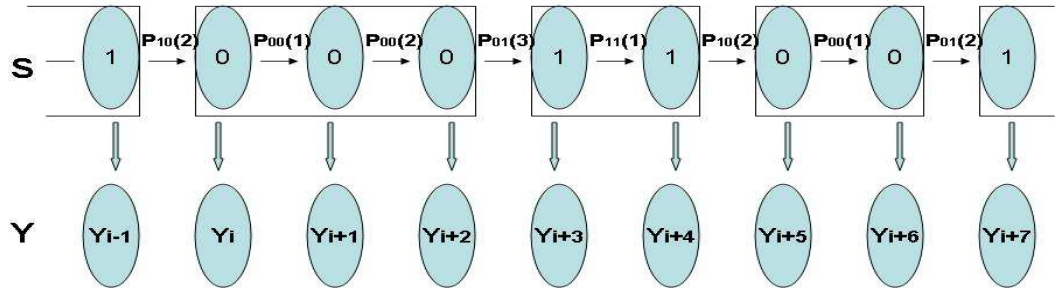

Figure 13: Representation of NSHMM

Let p_{ij} be the state transition probability in a VDHMM. Denote the state transition weight by $e_{ij}(d)$ and the state transition probability by $p_{ij}(d)$ in a NSHMM, where d is the state duration. If $d = 1$ or $p_{ij}(d)$ is constant over time, the model becomes a conventional HMM. If $p_{ij}(d)$, where $i \neq j$, is constant for all d and $p_{ii}(d)$ is replaced by a variable duration probability distribution without self-transitions, then this model becomes a VDHMM. Djric and Chun (2002) showed that NSHMM is equivalent to VDHMM when e_{ij}, the state transition weight is constant and set to be equal to p_{ij}. In general, NSHMM is more convenient for use, because the description of its data generation seems more natural, and it is more tractable for analysis (Djric and Chun, 2002). In addition, the implementation of MCMC sampling for estimation of the parameters and the states of the model is then much easier.

In this chapter, we develop a model with a different specified state duration distribution for each of the two states in the NSHMM frame, linker and non-linker. In a NSHMM frame, a NSHMM with a geometric duration distribution assuming constant transition probability is equivalent to a conventional HMM. We assume that protein sequence data is produced by a hidden Markov model and compositional variation is likely to reflect functional or structural differences between regions. Each region is classified into one of a finite number of states (e.g. linker region and non-linker region)

and we want to estimate the states given the observed protein sequence. We recognize the protein sequence data as continuous data instead of categorical data, which differs from the existing HMMs in computational protein sequence analysis. The observations in the HMMs for protein structure prediction are recognized as strings of amino acids (categorical variables), forming the primary sequence of a protein. Therefore, it is also important to find values which identify differences between linker regions and non-linker regions in a protein sequence. It is of value to obtain a probability for each residue being in the linker region, so that researchers can have better understanding of the protein sequence structure. This is the most advantageous property of our method over other methods that rely on amino acid propensity. The existing methods (Suyama and Ohara, 2003; Miyazaki et al., 2002) do not give probabilistic output.

Parameter estimation in HMMs usually relies on maximum likelihood estimation with dynamic programming-based algoritms or the Bayesian approach. In the Bayesian approach, we consider the HMM as a mixture model with missing data. We can associate observation y_i with a missing data z_i which represents the state from which y_i is generated. The EM algorithm was originally tailored for missing data structures, but the dependency between the states adds problems to the EM algorithm for mixture estimation. While the simulation of the missing data is straightforward for an independent structure, it is quite difficult to simulate from the distribution of missing data conditional on the observed data in HMMs. The use of a recurrent forward-backward formula, which is widespread in the literature for estimating HMM parameters, is time consuming and numerically sensitive. Instead, we employ an efficient Bayesian estimation of the model through MCMC methods, particularly Gibbs sampling, to implement inferences. Gibbs sampling effectively reduces the problem of sampling from a high-dimensional distribution to sampling from

a series of low-dimensional distributions.

We construct a representative dataset of multidomain protein sequences from the Pfam database release 14 (Bateman et al., 2002)(see section 3.2 for detail). We apply this model to the LD dataset which is the smoothed linker index data of protein sequences using linker index y^1 because the other linker indices give similar results, as we have seen in chapter II.

4.2 Model

Let $Y = (y_1, y_2, \cdots, y_n)'$ be a protein sequence generated by the corresponding hidden state $S = (s_1, s_2, \cdots, s_n)'$, where, y_i is the smoothed linker index of the i^{th} amino acid in a protein sequence with state s_i. Let the set of likelihood distribution parameters be θ,the set of duration distribution parameters be η and the set of state transition probability parameters τ. We say y_i is from a linker region if $s_i = 0$ and y_i is from a non-linker region if $s_i = 1$ in a protein sequence.

$$s_i = \begin{cases} 1 & \text{if } y_i \in \text{Non-linker region} \\ 0 & \text{if } y_i \in \text{Linker region} \end{cases}$$

Let $Q = (q_1, \cdots, q_m)'$ be the vector of the positions denoting the ends of each structural segment (state) with $q_0 = 0$ and m be the number of the segments in a protein sequence. The state duration variable, $d_i = q_i - q_{i-1}$, counts the number of times in which s_i remains in the same state.

$$d_i = \begin{cases} d_{i-1} + 1 & \text{if } s_i = s_{i-1} \\ 1 & \text{if } s_i \neq s_{i-1} \end{cases}$$

$D = (d_1, \cdots, d_m)'$ is the vector of the state durations. Note that d_i is also the number of observations (the length) in segment i.

The segments of data are summarized in Table 8.

Table 8: The segment of data

Q (position)	1,	$\cdots$,	q_1	q_1+1,	$\cdots$,	q_2	$\cdots$	$q_{m-1}+1$,	$\cdots$,	q_m
Y (observation)	y_1,	$\cdots$,	y_{q_1}	y_{q_1+1},	$\cdots$,	y_{q_2}	$\cdots$	$y_{q_{m-1}+1}$	$\cdots$	y_{q_m}
S (state)	s_1,	$\cdots$,	s_{q_1}	s_{q_1+1},	$\cdots$,	s_{q_2}	$\cdots$	$s_{q_{m-1}+1}$,	$\cdots$	s_{q_m}
D(duration)	1,	$\cdots$,	d_1	1,	$\cdots$,	d_2	$\cdots$	1,	$\cdots$	d_m
M (segment)		1			2		$\cdots$		m	

$\mathbf{P} = \{p_{s,jk}\}$ denotes the state transition probability matrix (between segments) and is defined as $p_{s,jk} = p(s_{q_i} = k|s_{q_{i-1}} = j)$. The transition probability (between observations) is defined as $p_{ik} = p(s_j = k|s_{j-1} = i)$. (See Figure 14 for notations)

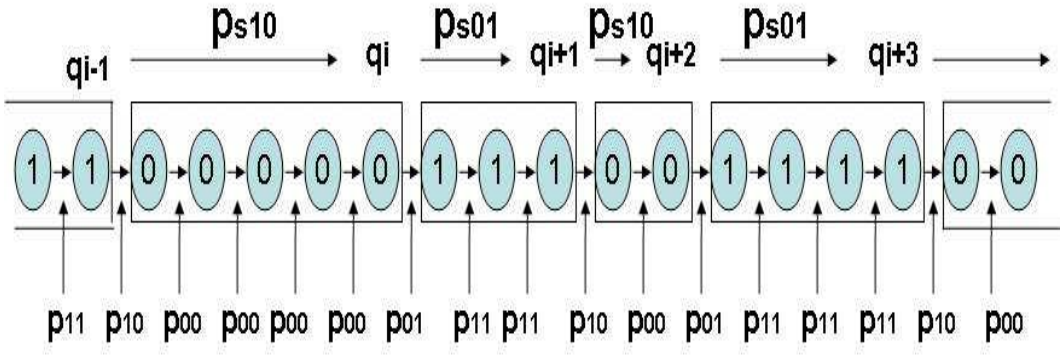

Figure 14: State transition probability and transition probability

The problem is to infer the values of (S, θ, η, τ) from the conditional joint posterior distributions $P(S, \theta, \eta, \tau|Y)$. The conditional joint posterior distribution is proportional to the joint distribution as follows.

$$\begin{aligned} P(S, \theta, \eta, \tau|Y) &\propto P(Y, S, \theta, \eta, \tau) \\ &\propto P(Y, S|\theta, \eta, \tau) \times P(\theta, \eta, \tau) \\ &\propto P(Y|S, \theta, \eta, \tau) \times P(S|\theta, \eta, \tau) \times P(\theta, \eta, \tau) \end{aligned}$$

We assume that the data Y are normally distributed and an observation y_i is conditionally independent from other observations within given segment i and also

from observations in other segments. The observations in the ith segment are denoted $y_{(q_{i-1},q_i]} = (y_{q_{i-1}+1}, \cdots, y_{q_i})$ and have the same state $s_{q_{i-1}+1} = \cdots = s_{q_i}$ by definition.

Both the mean and the variance of the data are parameterized in terms of a hidden state variable with Markov process. We assume the following model for the smoothed i^{th} linker index in a protein sequence.

$$\begin{aligned} y_i &= \mu_0 + \mu_1 s_i + (1 + \omega s_i)^{1/2} \epsilon_i \\ &\text{Restriction : } \ \mu_1 > 0 \end{aligned}$$

where, $\epsilon_i \sim N(0, \sigma^2)$ and ω denotes the proportional variance increase when $s_i = 1$. By the definition of linker index, it is reasonable to give the restriction that the mean linker index of linker region (μ_0) is smaller than the mean linker index of non-linker region ($\mu_0 + \mu_1$), because the linker index of a amino acid would be negative if it is more prevalent in the linker region than the other regions. The likelihood distribution for segment i given state j, θ_j, η_j and τ is

$$P(Y_{(q_{i-1},q_i]} | s_{q_{i-1}+1} = j, \theta_j, \eta_j, \tau) = \prod_{k=1}^{d_i} N\left(\mu_0 + \mu_1 s_k,\ (1 + \omega s_k)\sigma^2 \mid s_k = j, \theta_j, \eta_j, \tau\right)$$

where θ_j represents the parameters of the likelihood distribution whose state is j and η_j are the parameters of the duration distribution whose state is j. Therefore, the likelihood distribution of Y given S, θ, η and τ is

$$P(Y|S, \theta, \eta, \tau) = \prod_{i=1}^{m} P(Y_{(q_{i-1},q_i]} | s_{q_{i-1}+1} = j, \theta_j, \eta_j) = N((\underline{1},\ \underline{S})\underline{\mu},\ \sigma^2\Sigma | S, \theta, \eta, \tau)$$

where, $\underline{\mu} = (\mu_0, \mu_1)'$, $\underline{1} = (1, \cdots, 1)'$ is a $n \times 1$ vector and $\Sigma = diag((1 + ws_1), (1 + ws_2), \cdots, (1 + ws_n))$.

We can factor $P(S|\theta, \eta, \tau)$ as follows:

$$P(S|\theta, \eta, \tau) = \prod_{i=1}^{m} P(d | s_{q_{i-1}+1} = j, \theta, \eta, \tau) \times P(s_{q_i} | s_{q_{i-1}}, \theta, \eta, \tau)$$

The state transition probability distribution $P(s_{q_i}|s_{q_{i-1}}, \theta, \eta, \tau)$ is given by the following because there are two states.

$$\begin{pmatrix} p_{s,00} & p_{s,01} \\ p_{s,10} & p_{s,11} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

For simplicity, we suppress τ because the state transition probability matrix is given as shown above.

We assume that each state has a truncated Poisson duration distribution with a different parameter.

$$P(d|s_{q_i} = j, \theta_j, \eta_j) = \frac{\lambda_j^d e^{-\lambda_j}}{d!} \qquad d = 1, \cdots D_j$$

where D_j can be set to a reasonably large number for each hidden state and $j \in \{0, 1\}$. Numerous studies, including that of George and Heringa (2002a) show that the distributions of length of linker and non-linker region are significantly different. We incorporate the informative characteristic of length of the linker/non-linker regions into our model by using the specified state duration distribution.

The parameterization of the state duration is achieved by allowing all the transition probabilities $p_{ik} = p(s_j = k|s_{j-1} = i)$ to be functions of the duration time d.

$$p_{ik}(d) = P(s_t = k|s_{t-1} = s_{t-2} = \cdots = s_{t-d} = i)$$

The transition probabilities depend on state duration and this is why we refer to these HMMs as non-stationary HMMs. Djric and Chun (2002) show the relationship between the duration distribution and the transition probabilities $p_{ij}(d)$. The proof

of this formula is provided in appendix C.

$$p_{ii}(d) = \begin{cases} 1 - P(d|s_{t-1} = i) & d = 1 \\ \frac{1-\sum_{k=1}^{d} P(k|s_{t-1}=i)}{1-\sum_{k=1}^{d-1} P(k|s_{t-1}=i)} & d > 1 \end{cases} \tag{4.1}$$

$$p_{ij}(d) = \begin{cases} e_{ij}(d)P(d|s_{t-1} = i) & d = 1 \\ e_{ij}(d)\frac{P(d|s_{t-1}=i)}{1-\sum_{k=1}^{d-1} P(k|s_{t-1}=i)} & d > 1, \quad \text{where } i \neq j \end{cases} \tag{4.2}$$

The outward transition probability, $p_{ij}(d)$, can be obtained from $e_{ij}(d)(1-p_{ii}(d))$ using (4.1), where $e_{ij}(d)$ is the transition weight for the state j from i and the weights have to satisfy $\sum_{j=1,j\neq i} e_{ij}(d) = 1$ for all i and all d. However under the two state assumption, the transition weights become one for all d.

If we assume a geometric distribution with a different parameter for the state duration distribution as in (4.3) and also assume that the transition probability does not depend on the duration time d ($p_{jj}(d) = p_{jj}$) then the Non-stationary HMM becomes a conventional HMM.

$$P(d|s_{q_i} = j, \theta_j, \eta_j) = p_{jj}^{d-1}(1 - p_{jj}) \quad d = 1, \cdots D_j \tag{4.3}$$

where, $0 < p_{jj} < 1$ and $j \in \{0, 1\}$. We need to specify the prior distribution $P(\theta, \eta)$ to complete the joint distribution $P(Y, S, \theta, \eta)$.

The primary distribution of interest $P(S|Y)$ can be obtained by integrating out the conditional joint posterior distributions of $P(S, \theta, \eta|Y)$ with respect to θ.

$$\begin{aligned} P(S|Y) &= \int_\theta \int_\eta P(S, \theta, \eta|Y)\, d\theta\, d\eta \\ &\propto \int_\theta \int_\eta P(Y|S, \theta, \eta)P(S, \theta, \eta)\, d\theta\, d\eta \\ &\propto \int_\theta \int_\eta P(Y|S, \theta, \eta)P(S|\theta, \eta)P(\theta, \eta)\, d\theta\, d\eta \end{aligned} \tag{4.4}$$

Using equation (4.4), we can calculate the probability of being in a state k for

each amino acid i in a protein sequence given $y_i, s_{i-1} = l, , d_{i-1} = d, \theta$ and η.

$$\begin{aligned} P(s_i = k \quad | \quad & y_i, s_{i-1} = l, d_{i-1} = d, \theta, \eta) \\ = \quad & \frac{P(y_i|s_i = k, s_{i-1} = l, d_{i-1} = d, \theta, \eta)P(s_i = k|s_{i-1} = l, d_{i-1} = d, \theta, \eta)}{P(y_i|s_{i-1} = l, d_{i-1} = d, \theta, \eta)} \\ = \quad & \frac{P(y_i|s_i = k, d_{i-1} = d, \theta, \eta)p_{lk}(d)}{P(y_i|\theta, \eta)} \quad \because \ y_i \text{ dosen't depend on } s_{i-1} \text{ and } d_{i-1} \\ = \quad & \frac{P(y_i|s_i = k, d_{i-1} = d, \theta, \eta)p_{lk}(d)}{\sum_{j=0}^{1} P(y_i, s_i = j, d_{i-1} = d|\theta, \eta)} \\ = \quad & \frac{P(y_i|s_i = k, d_{i-1} = d, \theta, \eta)p_{lk}(d)}{\sum_{j=0}^{1} P(y_i|s_i = j, d_{i-1} = d, \theta, \eta)p_{lj}(d)} \end{aligned} \tag{4.5}$$

Once, these simulated sample values have been obtained from equation (4.5), any posterior moment or marginal distribution can be easily estimated. Specifically the posterior expectation can be estimated by the sample average, using equation (4.6).

$$\begin{aligned} E[P(s_i = k \quad | \quad & y_i, s_{i-1} = l, d_{i-1} = d, \theta, \eta)] \\ & = \frac{1}{m} \sum_{t=1}^{m} \frac{P(y_i|s_i^{(t)} = k, d_{i-1} = d, \theta^{(t)}, \eta^{(t)})p_{lk}^{(t)}(d)}{\sum_{j=0}^{1} P(y_i|s_i^{(t)} = j, d_{i-1} = d, \theta^{(t)}, \eta^{(t)}))p_{lj}^{(t)}(d)} \end{aligned} \tag{4.6}$$

where, t denotes the iteration in MCMC sampler, $k \in \{0, 1\}$ and m is the number of the MCMC samples from the posterior distribution after burn-in. We predict the state of an amino acid using the classification variable CV_i.

$$CV_i = \begin{cases} 1 & \text{if } \ E[P(s_i = k|y_i, s_{i-1} = l, d_{i-1} = d)] \leq 0.5 \\ 0 & \text{if } \ E[P(s_i = k|y_i, s_{i-1} = l, d_{i-1} = d)] > 0.5 \end{cases} \tag{4.7}$$

4.2.1 The Prior Distributions

The Bayesian approach to inference requires specification of a prior distribution for the parameters of the model. We assign mutually independent prior distributions for $\underline{\mu}$ and σ^2. The prior of $\underline{\mu}$ is assigned to be the conjugated normal distribution and the prior of σ^2 is the inverse gamma distribution. Here a random variable X is said

to follow Inverse Gamma distribution if $IG(\frac{a}{2}, \frac{2}{b}) \sim (\frac{1}{X})^{\frac{a}{2}+1} exp(-\frac{b}{2X})$.

$$\begin{aligned} \mu &\sim N\left(\begin{pmatrix}\mu_{0a}\\ \mu_{1a}\end{pmatrix}, \begin{pmatrix}\xi_{0a}, & 0\\ 0 & \xi_{1a}\end{pmatrix}\right) \\ \sigma^2 &\sim IG(\frac{a}{2}, \frac{2}{b}) \end{aligned}$$

Given hidden state s_i, ω only depends on the observations for $s_i = 1$. We use the expression $\bar{\omega} = (\omega + 1)$ in Albert and Chib (1993) to make ω represent the proportionate increase in variance when $s_i = 1$. Let the prior distribution of $\bar{\omega}$ be the truncated inverse gamma distribution.

$$\bar{\omega} \sim IG(\frac{a_w}{2}, \frac{2}{b_w}) \times I(\bar{\omega} > 1)$$

It is obvious that we only need to consider $(\lambda_0, \lambda_1)|S$ because λ_0, λ_1 are independent of (Y, θ) given S. We assign the conjugated gamma distributions for the priors of λ_j in the truncated Poisson distribution for each state.

$$P(\lambda_j) \propto \lambda_j^{a_j - 1} e^{-\lambda_j / b_j} \quad j \in \{0, 1\}$$

where a_j and b_j are positive.

4.3 Computation

The Bayesian inference is based on the posterior distribution. The quantities of interest are calculated by integrating the model parameters over the joint posterior distribution. Accurate approximation of these integrals can be made by the use of Markov Chain Monte Carlo (MCMC) methods. The posterior distribution is not available in explicit form so we use the MCMC method, specifically Gibbs sampling, to simulate the unknown parameters from the posterior distribution. The derivations of the full conditional distributions are provided in the appendix. Here $\theta|\cdot$ denotes θ conditioning on all other parameters.

It is convenient to transform data using $q_i = (1+\omega s_i)^{-1/2}$ so that the transformed data have constant variances, instead of variances that depend on the state.

$$q_i y_i \sim N((q_i, q_i s_i)\underline{\mu},\ \sigma^2) \equiv y_i^* \sim N(w_i^* \underline{\mu},\ \sigma^2)$$

Define $Y^* = (y_1^*, \ldots, y_n^*)'$ and $W^* = (w_1^*, \ldots, w_n^*)'$.

The full conditional distributions of $\underline{\mu}$ and σ^2 are as follows.

$$\begin{aligned} \mu|\cdot &\sim N\left(\mathbf{A}^{-1}(\sigma^{-2}W^{*'}Y^* + \mathbf{V}^{-1}\underline{\mu}_0),\quad \mathbf{A}^{-1}\right) \times I(\mu_1 > 0) \\ \sigma^2|\cdot &\sim IG\left(\frac{a+n}{2}, \frac{2}{b + (Y^* - W^*\underline{\mu})'(Y^* - W^*\underline{\mu})}\right) \end{aligned}$$

where $\mathbf{A} = (\mathbf{V}^{-1} + \sigma^{-2}W^{*'}W^*)$, $\underline{\mu}_0 = (\mu_{0a}, \mu_{1a})'$ and $\mathbf{V} = diag(\xi_{0a}, \xi_{1a})$.

The full conditional distribution of $\bar{\omega}$ is as follows.

$$\bar{\omega}|\cdot \sim IG\left(\frac{n_1 + a_w}{2}, \frac{2}{\sum_{t\in J}\left[\frac{(y_t - \mu_0 - \mu_1 s_t)^2}{\sigma}^2\right] + b_w}\right) \times I(\bar{\omega} > 1)$$

where $J = \{t|s_t = 1\}, t = 1, \cdots, n$ and n_1 is the number of observations whose state is 1.

The full conditional distributions of $\eta = (\lambda_0, \lambda_1)$ in the truncated Poisson distribution is

$$\lambda_j|\cdot \sim G(\bar{d}_j + a_j, (m_j + \frac{1}{b_j})^{-1}) \quad j \in \{0, 1\} \tag{4.8}$$

where $\bar{d}_j$ is the number of y_i's whose state is j and m_j is the number of segments whose state is j at the previous iteration. After we draw λ_j, we calculate $p_{ii}(d)$ and $p_{ij}(d)$, where $i \neq j$ using (4.1) and (4.2) respectively. Note that $P(d|s_t = j)$ in (4.1) and (4.2) is a truncated Poisson distribution with parameter λ_j at each iteration.

The full conditional distribution of s_i, $(i = 2, \cdots, n-1)$ at iteration k with

$p(s_1 = 1) = 1$ and $p(s_n = 1) = 1$ is

$$\begin{aligned} P(s_i|Y, S^{-i}, \theta, \eta) \quad \propto \quad & p_{s_{i-1}^{(k)} s_i}(d(s_{i-1}^{(k)})) \times p_{s_i s_{i+1}^{(k-1)}}(d(s_i)) \times p_{s_{i+1}^{(k-1)} s_{i+2}^{(k-1)}}(d(s_{i+1}^{(k-1)})) \\ & \times p_{s_{i+2}^{(k-1)} s_{i+3}^{(k-1)}}(d(s_{i+2}^{(k-1)})) \times \cdots \times p_{s_{i+\tau-1}^{(k-1)} s_{i+\tau}^{(k-1)}}(d(s_{i+\tau-1}^{(k-1)})) \\ & \times p(y_i|s_i, \theta, \eta) \end{aligned} \tag{4.9}$$

where, $s_{i+1}^{(k-1)} = s_{i+\tau-1}^{(k-1)}$, $s_{i+\tau-1}^{(k-1)} \neq s_{i+\tau}^{(k-1)}$ and $\tau \geq 2$, and $S^{-i} = (s_1, \cdots, s_{i-1}, s_{i+1}, \cdots, s_n)'$.

Derivation >

$$\begin{aligned} P(s_i|Y, S_{-i}, \theta, \eta) \quad = \quad & \frac{P(s_i|Y_i, S_{-i}, \theta, \eta) f(y_{i+1}, \cdots, y_n|Y_i, S_{-i}, s_i, \theta, \eta)}{f(y_{i+1}, \cdots, y_n|Y_i, S_{-i}, \theta, \eta)} \\ = \quad & P(s_i|y_i, S^{-i}, \theta, \eta) \quad \because y_i \text{are} \quad \text{independent} \\ \propto \quad & P(s_i|s_{i-1}^{(k)}, d(s_{i-1}^{(k)}), s_{i+1}^{(k-1)}, \cdots, s_{i+\tau}^{(k-1)}, \theta, \eta, y_i) \\ \propto \quad & P(s_i|s_{i-1}^{(k)}, d(s_{i-1}^{(k)})) \times P(s_{i+1}^{(k-1)}|s_i, d(s_i)) \times P(s_{i+2}^{(k-1)}|s_{i+1}^{(k-1)}, d(s_{i+1}^{(k-1)})) \\ & \times P(s_{i+3}^{(k-1)}|s_{i+2}^{(k-1)}, d(s_{i+2}^{(k-1)})) \times \cdots \times P(s_{i+\tau}^{(k-1)}|s_{i+\tau-1}^{(k-1)}, d(s_{i+\tau-1}^{(k-1)})) \\ & \times p(y_i|s_i, \theta, \eta) \\ \propto \quad & p_{s_{i-1}^{(k)} s_i}(d(s_{i-1}^{(k)})) \times p_{s_i s_{i+1}^{(k-1)}}(d(s_i)) \times p_{s_{i+1}^{(k-1)} s_{i+2}^{(k-1)}}(d(s_{i+1}^{(k-1)})) \\ & \times p_{s_{i+2}^{(k-1)} s_{i+3}^{(k-1)}}(d(s_{i+2}^{(k-1)})) \times \cdots \times p_{s_{i+\tau-1}^{(k-1)} s_{i+\tau}^{(k-1)}}(d(s_{i+\tau-1}^{(k-1)})) \\ & \times p(y_i|s_i, \theta, \eta) \end{aligned}$$

Therefore, the result follows.

4.4 Results

We apply our method to protein sequence datasets which we construct from Pfam-A database using y^1 as described in section 3.2. We divide each dataset into the training dataset and the test dataset randomly with the ratio of 4:1. We train the model with the training dataset of 642 sequences and test the trained model with the test dataset of 160 sequences. We run MCMC sampler, particularly Gibbs sampling

with 40,000 iterations and 10,000 burn-in. The choice of hyper-parameters is based on the data and the problem at hand and our choices are as follows. We let hyper-parameters for $\underline{\mu}$ be the sample means of the training dataset for each state and give each a large enough variance, 10, respectively. We assume $E(\bar{\omega}) = 1.5$, $\text{var}(\bar{\omega}) = 10$ and $E(\sigma^2) = 0.1$, $\text{var}(\sigma^2) = 10$ and fix the hyper-parameters that way. Fix the hyper-parameters for the λ_0 and λ_1 to have large variance for the priors, say 10,000.

We calculated the probability of being in the linker region, $p(s_i = 0|y_i)$, for each residue i along a protein sequence using (4.3). We compare the model in this chapter with the model in chapter III. We denote the model in chapter III as Model I and the model in this chapter as Model II. Model I is equivalent to a conventional HMM and Model II is a Non-stationary HMM with the truncated Poisson duration distribution. We apply our method to the test dataset to predict the inter-domain linker region and we use a whole sequence in the prediction instead of part of it. In Figures, $* = 1$ denotes residues are in the non-linker region and $* = 0$ denotes residues are linker region.

Figure 15 shows a case with good prediction in both models, in which probabilities in the linker region are much higher than in other regions. However, Model II gives more precise predictions than Model I.

Although both models give high probabilities to the linker region, each can also give high probabilities to some regions that are not linker regions, but may have similar structure. Most linker regions have negative linker index values, but some non-linker regions also have this pattern. Figure 16 shows one of these cases. There are two regions with high probability, but there is only one linker region in the protein. However, the probability of the actual linker region is higher than that of the other putative linker region. Also Model II gives a more precise probability boundary and fewer putative linker regions.

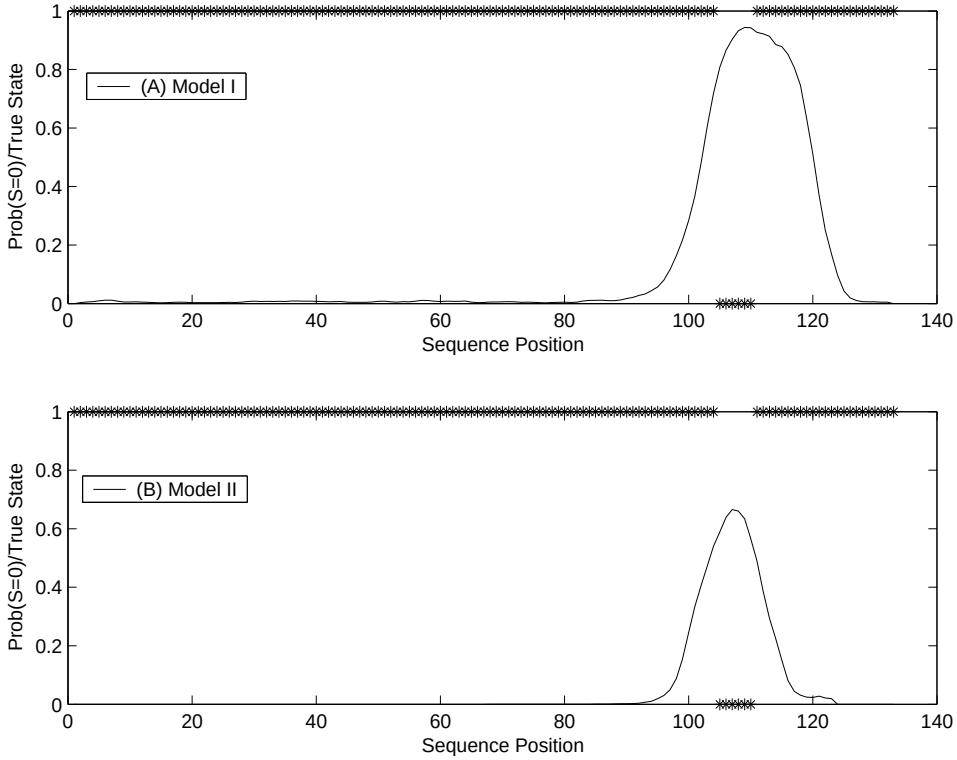

Figure 15: Probability of being in a linker region of ST-TREMBL Q7UD15

Figure 17(A) shows that the termini of the sequence can have high probabilities. The characteristics of the termini are not known. This case indicates that termini have patterns similar to those in the linker region. We have these patterns at the termini in Model I, because we do not utilize the length information in this model. This is a major weakness of using a geometric duration distribution. However, Model II can avoid this problem by incorporating the length of each region using the state duration distribution, even though it is not perfect as we see in Figure 17(B). This model starts with the non-linker state and uses length information, so that the putative region in the beginning of a sequence does not appear. We can find this pattern in all of the figures. Figure 17(B) shows a good result for those cases.

There are some protein sequences we can not predict well by using Model I. In

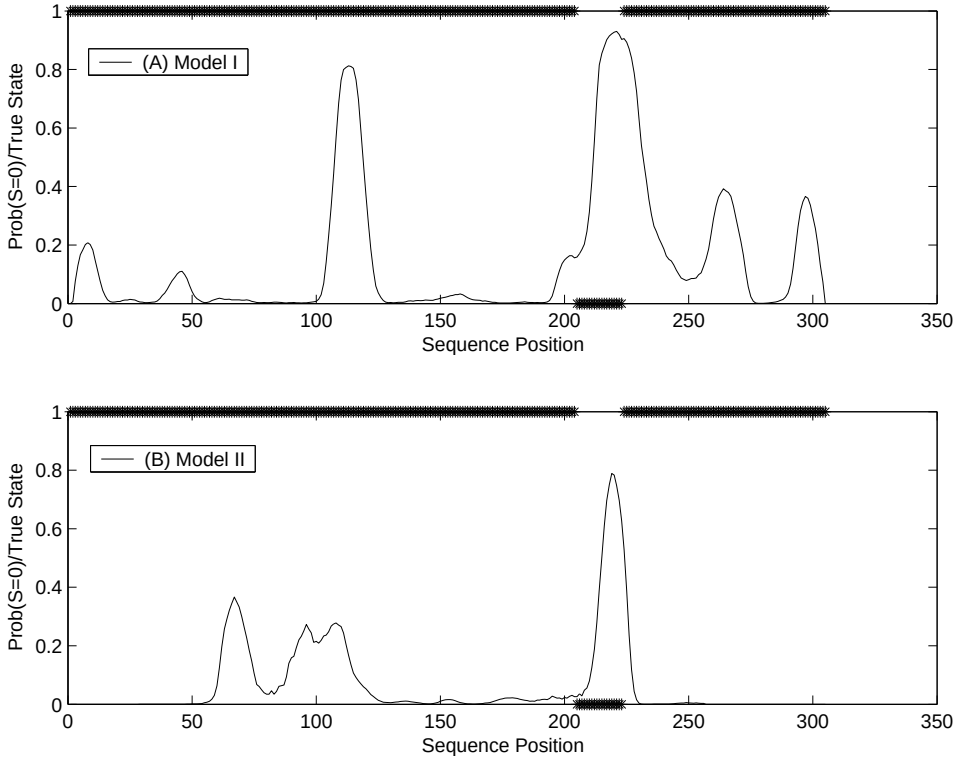

Figure 16: Probability of being in a linker region of ST-TREMBL Q89F20

Figure 18, Model II can give good results while Model I fails. In addition, Model II gives fewer incorrectly predicted putative linker regions than Model I. Model II tends to give higher probabilities in the linker region than Model I. Notice that both termini of a sequence tend to have high probability using Model I, but just the C terminus of a sequence has high probability when we use Model II.

Figure 19 shows a interesting case. Model I fails to predict the linker regions of this sequence as we see in Figure 19 (A). However, we can guess the linker region in Figure19 (B) because it has higher probability than the others even though it is not high enough to be recognized as a linker region.

We classify a residue into one of two states (linker or non-linker) using the classification variable (4.7), which is based on the probability in (4.6). To evaluate our

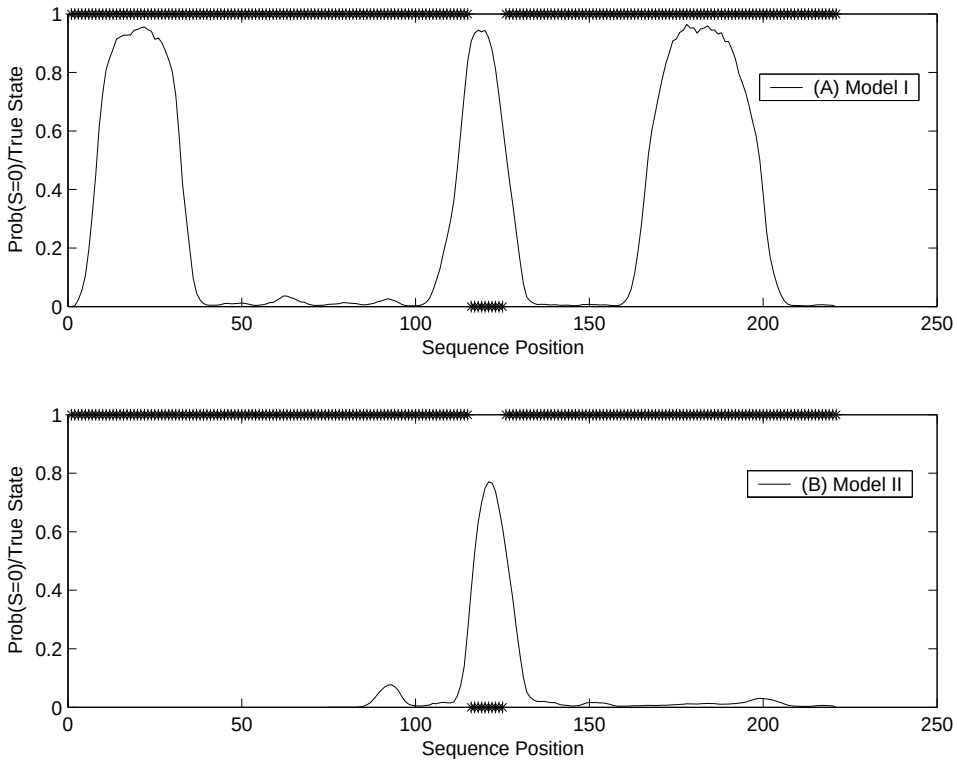

Figure 17: Probability of being in a linker region of ST-TREMBL Q7P6J3

method for the prediction of linker regions, sensitivity (Sn), specificity (Sp) and correlation coefficient (C) are reported with the test datasets. TP refers to those residues that are correctly labeled as linker and FP refers to residues that are labeled as linker while in fact they are non-linker. FN refers to residues that are labeled as non-linker while in fact they are linker. TN refers to residues that are correctly labeled as non-linker. The sensitivity is the percentage of actual linker residues that were predicted to be linker ($Sn = \frac{TP}{TP+FN}$) and the specificity is the percentage of predicted linker residues which are truly linker ($Sp = \frac{TP}{TP+FP}$) . The correlation coefficient (Matthews, 1975) is an indication of how much better a given prediction is than a random one $\left(C = \frac{(TP)(TN)-(FN)(FP)}{\sqrt{(TP+FN)(TP+FP)(TN+FN)(TN+FP)}}\right)$. $C = 1$ indicates perfect prediction while $C = 0$ is expected for a prediction no better than random. The evaluation results are

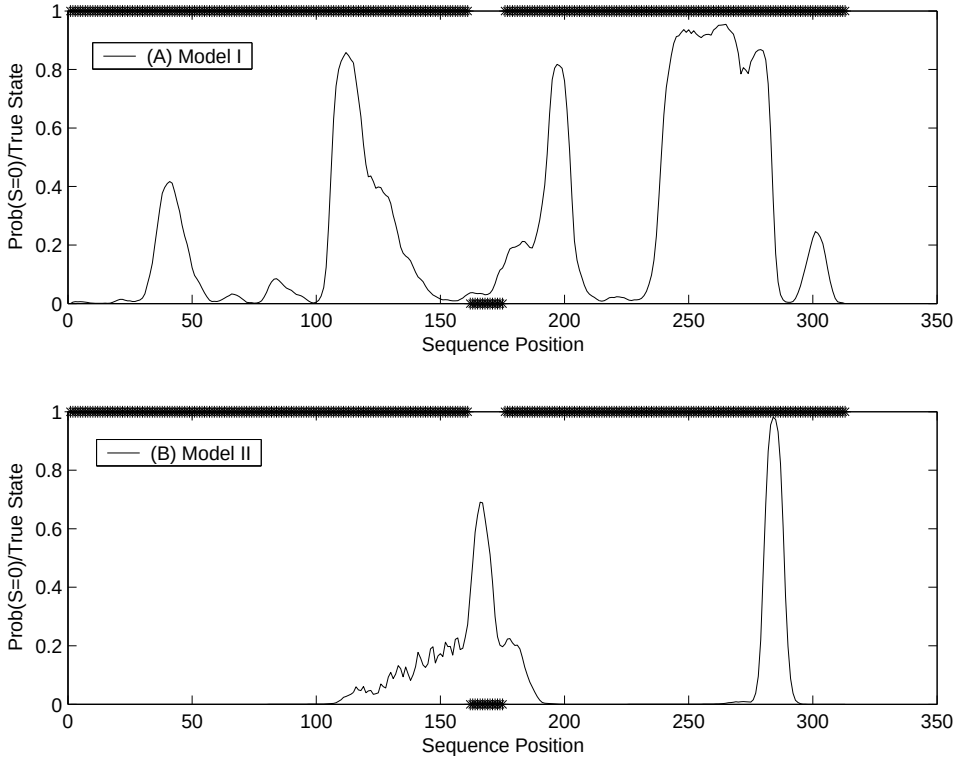

Figure 18: Probability of being in a linker region of ST-TREMBL Q81JL7

reported in Table 9.

Table 9: The comparison of models

	Sensitivity(Sn)	Specificity(Sp)	Correlation(C)
Model I	80.68	56.27	65.23
Model II	63.91	71.76	66.02

The sensitivity and the specificity of the stationary model are 80.68% and 56.27% respectively. Sensitivity and specificity of non-stationary model are 63.91% and 71.76% respectively. We can predict about 64% of the amino acids in the linker regions correctly out of all the amino acids in the linker regions using our method. Model I has better sensitivity, because Model I predicts a greater number of regions as linkers than Model II does (so it also gives many false positive). This is also the

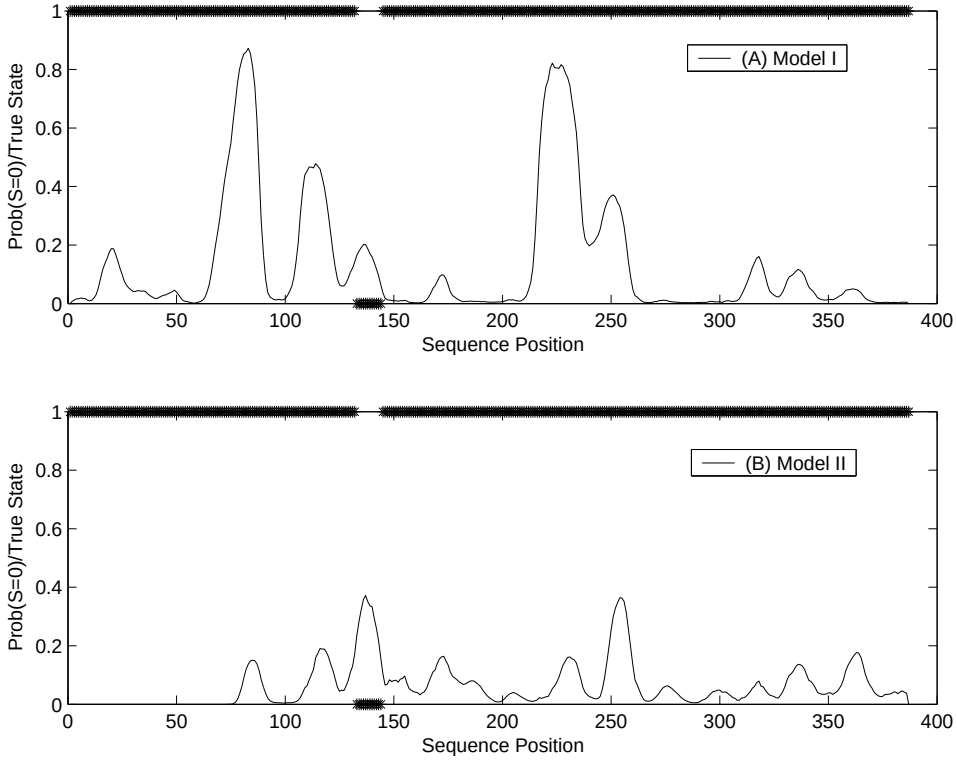

Figure 19: Probability of being in a linker region of ST-TREMBL Q7XXT5

reason Model I has low specificity. Model II gives a more precise determination of the boundary of linker regions, so that there are more linker residues which we fail to correctly identify. However, specificity is about 72%, which indicates that of all residues which the model predicts to be linkers, 72% are actually in the linker region. This indicates we have many fewer false positives than Model I. As we have seen in the figures, there are sequences for which the termini of the sequences have high probabilities in Model I. However, most of the high probabilities at the beginning terminus (N terminus) of a sequence are not exhibited in Model II. This is because we use a specified duration distribution. The standard Matthew's correlation coefficients (Matthews, 1975) are similar in both models. There are trade-offs between Model I and Model II, however Model II is preferable because it gives more accurate

probabilities in the linker region and fewer false predictions.

4.5 Discussion

We develop a model to detect the structurally different inter-domain linker and non-linker regions in a protein sequence by applying the compositional differences of amino acids between the two regions using a non-stationary hidden Markov model (NSHMM). We consider the duration probability density to utilize knowledge about the distribution of the lengths of the linker/non-linker regions in our model, because both the composition and the length of the linker region are important. In general, altering the length of linker regions connecting domains has been shown to affect protein stability, folding rates and domain-domain orientation (VanLeeuwen et al., 1997; Robinson and Sauer, 1998). A parameterization of the state duration is achieved by allowing all the transition probabilities to be functions of d, which is the duration of the same state stays.

We assume the independence of linker index data. This assumption makes sense, because we still do not understand the structure of domain and linker region well. However, we may consider the dependence of data in the model if we can identify the dependency between the data. We might consider that the observations follow a p^{th} order autoregressive model.

CHAPTER V

CONCLUSION

We develop Bayesian models to interpret biological data and use Markov chain Monte Carlo (MCMC) for the inference method.

Variable selection in cDNA microarray data highlights those genes which exhibit a different gene expression between two tissue types (e.g. normal and cancer) by removing redundant variables. We propose a two-level hierarchical Bayesian model for variable selection in cDNA data. We consider a multivariate Bayesian regression model and assign priors that favor sparseness in terms of number of variables (genes) used. We introduce the use of different priors to promote different degrees of sparseness, using a unified two-level hierarchical Bayesian model. We employ latent variables to specialize the model to a regression model. All the three models provide good performance in terms of gene selection, but the model based on the Jeffreys prior is preferable as there is no need to specify hyper-parameters or any type of threshold values. A formal choice of cut off value to select significant λ based on posterior or predictive criteria will be a topic of future research. Also, future research will consider the situation in which genes interact and extend the analysis to multi-category models.

We develop hidden Markov models to predict the linker regions in a protein using sequence information alone and use Markov chain Monte Carlo (MCMC) for the inference method. We apply our methods to a representative dataset of multidomain protein sequences which are constructed from the Pfam database release 14 (Bateman et al., 2002). We incorporate the differences in amino acid composition between different regions in a protein by using the linker index. Using the linker index, we rec-

ognize the protein sequence data as continuous data instead of categorical data, which is a different approach than the existing HMMs in computational protein sequence analysis. We also incorporate the duration probability density into HMMs to utilize prior information about the distribution of the lengths of the linker and non-linker regions. In general, altering the length of linker regions connecting domains has been shown to affect protein stability, folding rates and domain-domain orientation (Van-Leeuwen et al., 1997; Robinson and Sauer, 1998). An advantage of our methodology over the existing methods that do not give probabilistic output is that we may use the results to gain further insight into properties of the primary sequence. Furthermore, the probabilities can be considered when attempting to fragment unknown proteins into domains using molecular techniques prior to structure determination by NMR or X-ray crystallography. An advantage of both of our methods is that they can be applied to both the structural and evolutionary definitions of domain, depending on the dataset used, because neither relies on structure prediction. We assume the independence of linker index data. This assumption makes sense because we still do not understand the structure of domain and linker region well. However, we may consider dependence in the model if we can identify the dependency structure in the data. We might consider that the observations follow a p^{th} order autoregressive model. In protein sequence structure prediction, within each segment different states have different structure. For example, a linker region has less structure than a domain region, therefore a linker region would be less likely to have structured correlations between amino acids. Therefore, we may consider that the data follow a p_i^{th} order autoregressive model conditional on the hidden Markov chain state i. The work in this dissertation has demonstrated the value of applying a Bayesian approach with MCMC inference method to problems in bioinformatics. The approaches we have taken provide probabilities that biologists can use for further decision making.

REFERENCES

Albert, J. and Chib, S. (1993a). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association* **88**, 669–679.

Albert, J. and Chib, S. (1993b). Bayesian inference via gibbs sampling of autoregression time series subject to markov mean and variance shifts. *Journal of Business & econometric Statistics* **11**, 1–15.

Alexandrov, N. and Shindyalov, I. (2003). Pdp: protein domain parser. *Bioinformatics* **19**, 429–430.

Altman, R. B. and Raychaudhuri, S. (2001). Whole-genome expression analysis: challenges beyond clustering. *Curr. Opin. Biol.* **11**, 340–347.

Argos, P. (1990). An investigation of oligopeptides linking domains in protein tertiary structures and possible candidates for general gene fusion. *J. Mol. Biol.* **211**, 943–958.

Asai, K., Hayamizu, S., and Onizuka, K. (1993). Hmm with protein structure grammar. *Proceedings of the Hawaii International Conference on System Sciences* **1**, 783–791.

Bateman, A., Birney, E., Cerruti, L., Durbin, R., Etwiller, L., Eddy, S. R., Griffiths-Jones, S., Howe, K. L., Marshall, M., and Sonnhammer, E. L. L. (2002). The pfam protein families database. *Nucleic Acids Res.* **30**, 276–280.

Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M. C., Estreicher, A., Gasteiger, E., Martin, M. J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S., and Schnei-

der, M. (2003). The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic Acids Res.* **31**, 365–370.

Branden, C. and Tooze, J. (1999). *Introduction to Protein Structure*. New York: Garland Publishing, INC.

Brotherick, I., Robson, C. N., Browell, D. A., Shenfine, J., White, M. D., Cunliffe, W. J., Shenton, B. K., Egan, M., Webb, L. A., Lunt, L. G., Young, J. R., and Higgs, M. J. (1998). Cytokeratin expression in breast cancer : Phenotypic changes associated with disease progression.. *Cytometry* **32**, 301–308.

Burge, C. and Karlin, S. (1997). Prediction of complete gene structures in human. *J. Mol. Biol.* **268**, 78–94.

Campbell, C. (2002). Kernel methods: a survey of current techniques. *Neurocomputing* **48**, 63–84.

Cardon, L. R. and Stormo, G. D. (1992). Expectation maximization algorithm for identifying protein-binding sites with variable lengths from unaligned dna fragments. *J. Mol. Biol.* **223**, 159–170.

Churchill, G. A. (1989). Stochastic models for heterogeneous dna sequences. *Bull. Mathematical Biology* **51**, 79–94.

Churchill, G. A. and Lazareva, B. (1999). Bayesian restoration of a hidden markov chain with applications to dna sequencing. *Journal of Computational Biology* **6**, 261–277.

Devore, J. and Peck, R. (1997). *Statistics: The Exploration and Analysis of Data*. Pacific Grove, CA: Duxbury Press.

Djric, P. M. and Chun, J. H. (2002). An mcmc sampling approach to estimation of nonstationary hidden markov models. *IEEE Transactions on Signal Processing* **50**, 1113–1123.

Dudoit, S., Yang, Y., Callow, M., and Speed, T. (2000). Statistical methods for identifying differentially expressed genes in replicated cdna microarray experiments. Tech. rep. 578, University of California Berkeley.

Enright, A. J. and Ouzonis, C. A. (2000). Generage: a robust algorithm for sequence clustering and domain detection. *Bioinformatics* **16**, 451–457.

Ferguson, J. D. (1980). Variable duration models for speech. *Proceedings of the Symposium on the Application of Hidden Markov Models to Text and Speech* **1**, 143–179.

Figueiredo, M. A. T. (2001). Adaptive sparseness using jeffreys prior. *Advances in Neural Information Processing Systems* **14**, 697–704.

Galzitskaya, O. and Melnik, B. (2003). Prediction of protein domain boundaries from sequence alone. *Protein Sci.* **12**, 696–701.

Gelfand, A. and Smith, A. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association* **88**, 881–889.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2000). *Bayesian data analysis*. Boca Raton, FL: Chapman & Hall/CRC.

George, R. A. and Heringa, J. (2002a). An analysis of protein domain linkers: their classification and role in protein folding. *Protein Eng.* **15**, 871–879.

George, R. A. and Heringa, J. (2002b). Protein domain identification and improved sequence similarity searching using psi-blast. *Proteins* **48**, 672–681.

George, R. A. and Heringa, J. (2002c). Snapdragon: a method to delineate protein structural domains from sequence data. *J. Mol. Biol.* **316**, 839–851.

Gilks, W., Richardson, S., and Spiegelhalter, D. (1996). *Markov Chain Monte Carlo in Practice*. London: Chapman and Hall.

Gokhale, R. S. and Khosla, C. (2000). Role of linkers in communication between protein modules. *Curr. Opin. Chem. Biol.* **4**, 22–27.

Golub, T. R., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., and Lender, E. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* **286**, 531–537.

Gouzy, J., Eugene, P., Greene, E. A., Kahn, D., and Corpet, F. (1997). Xdom, a graphical tool to analyse domain rearrangements in any set of protein sequences. *Comput. Appl. Biosci.* **13**, 601–608.

Gracy, J. and Argos, P. (1998). Automated protein sequence database classification.ii. delineation of domain boundaries from sequence similarities. *Bioinformatics* **14**, 174–187.

Grant, G., Manduchi, E., and Stoeckert, C. (2002). Using non-parametric methods in the context of multiple testing to identify differentially expressed genes. Tech. rep. CAMDA00, Durham, NC, Duke University.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classifcation using support vector machines. *Machine Learning* **46**, 389–422.

Harlan, D. M., Graff, J. M., Stumpo, D. J., and Blackshear, P. J. (1991). The human myristoylated alanine-rich c kinae subtrate 9markcks gene (macs): analysis of

its gene product, promoter, and chromosal localization. *Journal of Biological Chemistry* **266**, 14399–14405.

Haussler, D., Krogh, A., Mian, I. S., and Sjolander, K. (1993). Protein modeling using hidden markov models: analysis of globins. *Proceedings of the Hawaii International Conference on System Sciences* **1**, 792–802.

Hellstrom, I., Beaumier, P. L., and Hellstrm, K. E. (1986). Antitumor effects of 16, an igg2a antibody that reacts with most human carcinomas. *Proc. Natl. Acad. Sci.U.S.A.* **83**, 7059–7063.

Hendenfalk, I., Duggan, D., Chen, Y., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., and Raffeld, M. (2001). Gene expression profiles in hereditary breast cancer. *The New England Journal of Medicine* **344**, 539–548.

Henderson, J., Salzberg, S., and Fasman, K. H. (1997). Finding genes in dna with a hidden markov model. *J. Comput. Biol.* **4**, 127–141.

Holm, L. and Sander, C. (1994). Parser for protein folding units. *Proteins* **19**, 256–268.

Islam, S. A., Luo, J., and Sternberg, M. J. (1995). Identification and analysis of domains in proteins. *Protein Eng.* **8**, 513–525.

Kamps, M. P., Murre, C., Sun, X. H., and Baltimore, D. (1990). A new homeobox gene contributes the dna binding domain of the t(1;19) translocation protein in pre-b all. *Cell* **6**, 547–555.

Kim, S., Dougherty, E. R., Barrera, J., Chen, Y., Bitter, M., and Trent, J. (2002). Strong feature sets from small sapmles c.j.. *Computational Biology* **7**, 673–679.

Kingsmore, S. F. (1995). Genetic mapping of the t lymphocyte-specific transcription factor 7 gene on mouse chromosome 11. *Mamm. Genome* **6**, 378.

Kulp, D., Haussler, D., Reese, M., and Eeckman, F. H. (1996). A generalized hidden markov model for the recognition of human genes in dna. *Proceedings of the International Conference Intellegent Systems* **4**, 134–142.

Lander, E. and Green, P. (1987). Construction of multilocus genetic linkage maps in human. *Proc. Natl. Acad. Sci. USA* **84**, 2363–2367.

Lee, K. E., Sha, N., Dougherty, E. R., Vanucci, M., and Mallick, B. K. (2003). Gene selection: a bayesian variable selection approach. *Bioinformatics* **19**, 90–97.

Li, Y., Campbell, C., and Tipping, M. (2002). Bayesian automatic relevance determination algorithms for classifying gene expression data. *Bioinformatics* **18**, 1332–1339.

Linding, R., Russell, R. B., Neduva, V., and Gibson, T. J. (2003). Globplot: exploring protein sequences for globularity and disorder. *Nucleic Acids Res.* **31**, 3701–3708.

Liu, J. and Rost, B. (2004a). Chop: parsing proteins into structural domains. *Nucleic Acids Res.* **32**, 569–571.

Liu, J. and Rost, B. (2004b). Sequence-based prediction of protein domains. *Nucleic Acids Res.* **32**, 3522–3530.

Marken, J. S., Schieven, G. L., Hellstrm, I., Hellstrm, K. E., and Aruffo, A. (1992). Cloning and expression of the tumor associated antigen. *Proc. Natl. Acad. Sci.U.S.A.* **89**, 3503–3507.

Marsden, R. L., McGuffin, L. J., and Jones, D. T. (2002). Rapid protein domain assignment from amino acid sequence using predicted secondary structure. *Protein Sci.* **11**, 2814–2824.

Matsuda, S., Kawamura-Tsuzuku, J., Ohsugi, M., Yoshida, M., Emi, M., Nakamura, Y., Onda, M., Yoshida, Y., Nishiyama, A., and Yamamoto, T. (1996). Tob, a nobel protein that interacts with p185erbb2, is associated with antiproliferative activity. *Oncogene* **12**, 705–713.

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochi. Biophys. Acta* **405**, 442–451.

Michael, J. R., Schucany, W. R., and Haas, R. W. (1976). Generating random variates using transformations with multiple roots. *American Statistician* **30**, 88–90.

Miranker, A. D. (2000). Protein complexes and analysis of their assembly by mass spectrometry. *Curr. Opin. Struct. Biol.* **10**, 106–606.

Miyazaki, S., Kuroda, Y., and Yokoyama, S. (2002). Characterization and prediction of linker sequences of multidomain proteins by a neural network. *J. Struct. Funct. Genomics* **2**, 37–51.

Murvai, J., Vlahovicek, K., Szepesvari, C., and Pongor, S. (2001). Prediction of protein functional domains from sequences using artificial neural networks. *Genome Res.* **11**, 1410–1417.

Osake, M., Rowley, J. D., and Zeleznik-Le, N. J. (1999). Msf, a fusion partner gene of mll, in a theraphy-related acute myeloid leukemia with at(11:17). *Proc. Natl. Acad. Sci. USA* **96**, 6428–6433.

Pan, W. (2002). A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments. *Bioinformatics* **18**, 546–554.

Pearson, W. R. and Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA* **85**, 2444–2448.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proc. IEEE.* **77**, 257–286.

Robert, C. (1999). Simulation of truncated normal variables. *Statistics and Computing* **5**, 121–125.

Robert, C. P. and Mengersen, K. L. (1999). Reparameterisation issues in mixture modeling and their bearing on mcmc algorithms. *Computational Statistics & Data Analysis* **29**, 325–343.

Robinson, C. R. and Sauer, R. T. (1998). Optimizing the stability of single-chain proteins by linker length and composition mutagenesis. *Proc. Natl. Acad. Sci. USA* **95**, 5929–5934.

Schmidler, S., Liu, J. S., and Brutlag, D. L. (2000). Bayesian segmentation of protein secondary structure. *J. Comput. Biol.* **7**, 233–248.

Schmidler, S., Liu, J. S., and Brutlag, D. L. (2001). Bayesian protein structure prediction. *Case Studies in Bayesian Statistics* **5**, 363–378.

Siddiqui, A. S. and Barton, G. J. (1995). Continuous and discontinuous domains: an algorithm for the automatic generation of reliable protein domain definitions. *Protein Sci.* **4**, 872–884.

Sin, B. and Kim, J. (1995). Nonstationary hidden markov models. *Signal Procss.* **46**, 31–46.

Smith, A., Satagopan, J., Gonen, M., and Begg, C. (2002). Exploring class prediction for leukemia gene expression data. Tech. rep. CAMDA02, Durham, NC, Duke University.

Sonnhammer, E. L. L. and Kahn, D. (1994). Modular arrangement of proteins as inferred from analysis of homology. *Protein Sci.* **3**, 482–492.

Suyama, M. and Ohara, O. (2003). Domcut: prediction of inter-domain linker regions in amino acid sequences. *Bioinformatics* **19**, 673–674.

Swindells, M. B. (1995). A procedure for detecting structural domains in proteins. *Protein Sci.* **4**, 103–112.

Tanaka, T., Kuroda, Y., and Yokoyama, S. (2003). Characteristics and prediction of domain linker sequences in multidomain proteins. *J. Struct. Funct. Genomics* **4**, 79–85.

Taylor, W. R. (1999). Protein structural domain identification. *Protein Eng.* **12**, 203–216.

Thomas, J. G., Olson, J. M., Tapscott, S. J., and Zhao, L. P. (2001). An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles. *Genome Research* **11**, 1227–1236.

Thorsteinsdottir, U., Krosl, J., Hoang, T., and Sauvageau, G. (1999). The oncoprotein e2a-pbx collaborates with hoax2 to acutely transform primary bone mallow cells. *Molecular Cell Biology* **19**, 6355–6366.

Tipping, M. E. (2000). The relevance vector machine. *Advances in Neural Information Processing Systems* **12**, 652–658.

Troyanskaya, O. G., Garber, M. E., Brown, P. O., Botstein, D., and Altman, R. B. (2002). Nonparametric methods for identifying differentially expressed genes in microarray data. *Bioinformatics* **18**, 1361–1454.

Udwary, D. W., Merski, M., and Townsend, C. A. (2002). A method for prediction of linker regions within large multifunctional proteins, and its application to a type i polyketide synthase. *J. Mol. Biol.* **323**, 585–598.

VanLeeuwen, H., Strating, M. J., Rensen, M., Laat, W. D., and der Vliet, P. C. V. (1997). Linker length and composition influence the flexibility of oct-1 dna binding. *European Molecular Biology Organization Journal* **16**, 2043–2053.

Watson, J. D. and Crick, F. H. C. (1953). A structure for deoxyribose nucleic acid. *Nature* **171**, 737–739.

Wernisch, L., Hunting, M., and Wodak, S. J. (1999). Identification of structural domains in proteins by a graph heuristic. *Proteins* **35**, 338–352.

Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., and Vapnik, V. (2001). Feature selection for svms. *Advances in Neural Information Processing Systems* **13**, 668–674.

Wheelan, S. J., Marchler-Bauer, A., and Bryant, S. H. (2000). Domain size distributions can predict domain boundaries. *Bioinformatics* **16**, 697–701.

Williams, C. (1998). *Prediction with Gaussian processes: from linear regression to linear prediction and beyond.* Assinippi Park, NY: Kluwer Academic Press.

Williams, C. and Barber, D. (1998). Bayesian classification with gaussian priors. *TEEE Trans. on Pattern Analysis and machine Intelligence* **20**, 1342–1351.

Williams, P. (1995). Bayesian regularization and pruning using a laplace prior. *Newral Computation* **7**, 117–143.

Wootton, J. (1994). Non-globular domains in protein sequences: automated segmentation using complexity measures. *Comput. Chem.* **18**, 269–285.

Xu, Y., Xu, D., and Gabow, H. N. (2000). Protein domain decomposition using a graph-theoretic approach. *Bioinformatics* **16**, 1091–1104.

Yeoh, E. J., Ross, M. E., Shurtleff, S. A., Williams, W. K., Patel, D., Mahfouz, R., Behm, F. G., Raimondi, S. C., Relling, M. V., Patel, A., Cheng, C., Campana, D., Wilkins, D., Zhou, X., Li, J., Liu, H., Pui, C. H., Evans, W. E., Naeve, C., Wong, L., and Downing, J. R. (2002). Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. *Cancer Cell* **1**, 133–143.

APPENDIX A

The purpose of this appendix is to prove equation (2.1) and to provide the derivation of the full conditional distributions in Chapter II.

Proofs of equtaion 2.1

The Laplace prior can be expressed as a zero-mean Gaussian prior with an independent exponentially distributed variance :

$$\pi(\beta_i|\gamma) = \int_0^\infty \pi(\beta_i|\lambda_i)\pi(\lambda_i|\gamma)d\lambda_i \propto Laplace(0, \frac{1}{\sqrt{\gamma}}).$$

$< Proof >$

$$\begin{aligned}
\pi(\beta_i|\gamma) &= \int_0^\infty p(\beta_i|\lambda_i)p(\lambda_i|\gamma)d\lambda_i \\
&= \int_0^\infty \frac{1}{\sqrt{2\pi\lambda_i}} exp\left(-\frac{\beta_i^2}{2\lambda_i}\right)\frac{\gamma}{2} exp\left(-\frac{\gamma\lambda_i}{2}\right)d\lambda_i \\
&= \frac{\gamma}{2\sqrt{2\pi}}\int_0^\infty \frac{1}{\sqrt{\lambda_i}} exp\left(-\frac{|\beta_i|^2+\gamma\lambda_i^2}{2\lambda_i}\right)d\lambda_i \\
&= \frac{\gamma}{2}\int_0^\infty \frac{1}{\sqrt{2\pi\lambda_i}} exp\left(-\frac{(|\beta_i|-\sqrt{\gamma}\lambda_i)^2}{2\lambda_i}\right) exp(-|\beta_i|\sqrt{\gamma})d\lambda_i \\
&\propto \frac{\sqrt{\gamma}}{2} exp(-|\beta_i|\sqrt{\gamma})\int_0^\infty \frac{\sqrt{\gamma}}{\sqrt{2\pi\lambda_i}} exp\left(-\frac{\gamma(\lambda_i-\frac{|\beta_i|}{\sqrt{\gamma}})^2}{2\lambda_i}\right)d\lambda_i \\
&\propto \frac{\sqrt{\gamma}}{2} exp(-|\beta_i|\sqrt{\gamma})\int_0^\infty \sqrt{\frac{\gamma}{2\pi\lambda_i}} exp\left(-\frac{\gamma(\lambda_i-\frac{|\beta_i|}{\sqrt{\gamma}})^2}{2\lambda_i}\right)d\lambda_i \\
&\quad \text{Let}\sqrt{\lambda_i} = x_i,\ \ d\lambda_i = 2x_i dx_i \\
&\propto \frac{\sqrt{\gamma}}{2} exp(-|\beta_i|\sqrt{\gamma})\int_0^\infty exp\left(-\frac{\gamma x_i^2}{2}\right) exp\left(-\frac{\beta_i^2}{2x_i^2}\right)dx_i \\
&\propto \frac{\sqrt{\gamma}}{2} exp(-|\beta_i|\sqrt{\gamma})
\end{aligned}$$

Therefore, the result follows.

The full conditional distribution of β (2.2)

$$\begin{aligned}\pi(\beta|\mathbf{Z},\mathbf{y},\mathbf{\Lambda}) &\propto N((\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{X}'\mathbf{Z},(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}) \\ \text{where} \quad & (\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}=\mathbf{\Lambda}-\mathbf{\Lambda}\mathbf{X}'(\mathbf{X}\mathbf{\Lambda}\mathbf{X}'+\mathbf{I})^{-1}\mathbf{X}\mathbf{\Lambda}\end{aligned}$$

< Derivation >

$$\begin{aligned}\pi(Z,\beta,\mathbf{\Lambda}|Y) &\propto \pi(Z|\beta,\mathbf{\Lambda},Y)\times\pi(\beta|\mathbf{\Lambda},Y)\times\pi(\mathbf{\Lambda}|Y) \\ &\propto |2\pi I|^{-1/2}exp[-\frac{1}{2}(Z-\mathbf{X}\beta)'(Z-\mathbf{X}\beta)] \\ &\quad \times|2\pi\mathbf{\Lambda}|^{-1/2}exp[[-\frac{1}{2}\beta'\mathbf{\Lambda}\beta]\pi(\mathbf{\Lambda}|Y)\end{aligned}$$

$$\begin{aligned}\pi(\beta|Z\mathbf{\Lambda},Y) &\propto exp[-\frac{1}{2}\{\beta'(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})\beta-2\beta'\mathbf{X}'Z\}] \\ &\propto exp[-\frac{1}{2}\{(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})\beta\beta'-2\mathbf{X}'Z\beta'\}] \\ &\propto exp[[-\frac{(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})}{2}\{\beta-(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{X}'Z)\} \\ &\quad \{\beta-(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{X}'Z)\}'] \\ &\propto exp[-\frac{1}{2}(\beta-(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{X}'Z)'(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1}) \\ &\quad (\beta-(\mathbf{X}'\mathbf{X}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{X}'Z)]\end{aligned}$$

The full conditional distribution of $\mathbf{\Lambda}$ (2.3)

$$\pi(\mathbf{\Lambda}|\mathbf{Z},\mathbf{y},\beta) \propto \Pi_{i=1}^{p} IG(\frac{a+1}{2},\frac{2}{b+\beta_i^2})$$

< Derivation >

$$\begin{aligned}\pi(\mathbf{\Lambda}|Y) &= \prod_{i=1}^{p}(\lambda_i^{-1})^{a/2+1}exp[-\frac{b\lambda_i^{-1}}{2}] \\ \pi(\mathbf{\Lambda}|\mathbf{Z},\mathbf{y},\beta) &\propto |\mathbf{\Lambda}|^{-1/2}exp\left[-\frac{1}{2}\beta'\mathbf{\Lambda}^{-1}\beta\right]\prod_{i=1}^{p}(\lambda_i^{-1})^{\frac{a}{2}+1}exp\left[-\frac{b\lambda_i^{-1}}{2}\right] \\ &\propto \prod_{i=1}^{p}(\lambda_i^{-1})^{\frac{a+1}{2}-1}exp\left[-\frac{(b+\beta_i^2)\lambda_i^{-1}}{2}\right]\end{aligned}$$

The full conditional distribution of $\mathbf{\Lambda}$ (2.4)

$$\pi(\mathbf{\Lambda}^{-1}|\mathbf{Z},\mathbf{y},\beta) \quad \propto \quad \Pi_{i=1}^{p} InvGauss(\frac{\sqrt{\gamma}}{\beta_i},\gamma)$$

$$\text{where,} \quad InvGauss(\mu,\lambda) = \sqrt{\frac{\lambda}{2\pi x^3}} exp(-\frac{\lambda}{2\mu^2}\frac{(x-\mu)^2}{x}), x \geq 0$$

$< Derivation >$

$$\begin{aligned}
\pi(\mathbf{\Lambda}|Y) &= \prod_{i=1}^{p} \frac{\gamma}{2} exp\left(-\frac{\gamma\lambda_i}{2}\right) \\
\pi(\mathbf{\Lambda}|\mathbf{Z},\mathbf{y},\beta) &\propto |\mathbf{\Lambda}|^{-1/2} exp\left[-\frac{1}{2}\beta'\mathbf{\Lambda}^{-1}\beta\right] \prod_{i=1}^{p} \frac{\gamma}{2} exp\left(-\frac{\gamma\lambda_i}{2}\right) \\
\text{Let } \lambda^{-1} = \theta_i, \quad \text{then} \quad & \frac{d\lambda_i}{d\theta_i} = -\frac{1}{\theta_i^2} d\lambda_i, \text{ and } \theta = diag(\lambda_1^{-1},\cdots,\lambda_p^{-1}) \\
\pi(\theta|\mathbf{Z},\mathbf{y},\beta) &\propto \prod_{i=1}^{p} \frac{\gamma}{\sqrt{\frac{2\pi}{\theta_i}}} exp\left(-\frac{\beta_i^2\theta_i^2+\gamma}{2\theta_i}\right) \times \left|-\frac{1}{\theta_i^2}\right| \\
&\propto \prod_{i=1}^{p} \frac{\gamma}{\sqrt{2\pi\theta_i^3}} exp\left(-\frac{\beta_i^2\theta_i^2+\gamma}{2\theta_i}\right) \\
&\propto \prod_{i=1}^{p} \frac{\gamma}{\sqrt{2\pi\theta_i^3}} exp\left(-\frac{\theta_i^2+\gamma/\beta_i^2}{2\theta_i/\beta_i^2}\right) \\
&\propto \prod_{i=1}^{p} \frac{\gamma}{\sqrt{2\pi\theta_i^3}} exp\left(-\frac{(\theta_i-\sqrt{\gamma}/\beta_i)^2}{2\theta_i/\beta_i^2}\right) \\
&\propto \prod_{i=1}^{p} \frac{\gamma}{\sqrt{2\pi\theta_i^3}} exp\left(-\frac{\gamma}{2(\frac{\sqrt{\gamma}}{\beta_i})^2}\frac{(\theta_i-\sqrt{\gamma}/\beta_i)^2}{\theta_i}\right)
\end{aligned}$$

The full conditional distribution of $\mathbf{\Lambda}$ (2.5)

$$\pi(\mathbf{\Lambda}^{-1}|\mathbf{Z},\mathbf{y},\beta) \propto \Pi_{i=1}^{p} G(\frac{1}{2},\frac{2}{\beta_i^2})$$

$< Derivation >$

$$\begin{aligned}
\pi(\mathbf{\Lambda}|Y) &= \prod_{i=1}^{p}\frac{1}{\lambda_i} \\
\text{Let } \lambda^{-1}=\theta_i, \quad \text{then } \quad \frac{d\lambda_i}{d\theta_i} &= -\frac{1}{\theta_i^2}d\lambda_i, \text{ and } \theta = diag(\lambda_1^{-1},\cdots,\lambda_p^{-1}) \\
\pi(\theta|\mathbf{Z},\mathbf{y},\beta) &\propto |\theta|^{1/2} exp\left[-\frac{1}{2}\beta'\theta\beta\right]\prod_{i=1}^{p}\theta_i \times \left|-\frac{1}{\theta_i^2}\right| \\
&\propto \prod_{i=1}^{p}\theta_i^{\frac{3}{2}} exp(-\frac{\beta_i^2\theta_i}{2}) \times \left|-\frac{1}{\theta_i^2}\right| \quad \because \theta_i > 0 \\
&\propto \prod_{i=1}^{p}\theta_i^{\frac{1}{2}-1} exp(-\frac{\beta_i^2\theta_i}{2})
\end{aligned}$$

APPENDIX B

The purpose of this appendix is to provide derivations of full conditional distributions in Chapter III.

The full conditional distributions of $\underline{\mu}$ (3.5)

$$\underline{\mu}|\cdot \sim N\left(\mathbf{A}^{-1}(\sigma^{-2}Q^{*'}Y^{*} + \mathbf{V}^{-1}\underline{\mu}_0),\ \mathbf{A}^{-1}\right) \times I(\mu_1 > 0)$$

where, $A = (\mathbf{V}^{-1} + \sigma^{-2}Q^{*'}Q^{*})$ $\quad \underline{\mu}_0 = (\mu_{0a}, \mu_{1a})'$ and $\mathbf{V} = diag(\xi_{0a}, \xi_{1a})$.

Derivation >

$$\begin{aligned}
\underline{\mu}|\cdot \ &\propto\ exp\left[-\frac{1}{2\sigma^2}(Y^{*} - Q^{*}\underline{\mu})'(Y^{*} - Q^{*}\underline{\mu})\right] \times exp\left[-\frac{1}{2}(\underline{\mu} - \underline{\mu}_0)'\mathbf{V}^{-1}(\underline{\mu} - \underline{\mu}_0)\right] \\
&\propto\ exp\left[\frac{2\sigma^{-2}\underline{\mu}'Q^{*'}Y^{*} - \sigma^{-2}\underline{\mu}'Q^{*'}Q^{*}\underline{\mu} - \underline{\mu}'\mathbf{V}^{-1}\underline{\mu} + 2\underline{\mu}'\mathbf{V}^{-1}\underline{\mu}_0}{2}\right] \\
&\propto\ exp\left[-\frac{1}{2}\{\underline{\mu}'(\mathbf{V}^{-1} + \sigma^{-2}Q^{*'}Q^{*})\underline{\mu} - 2\underline{\mu}'(\sigma^{-2}Q^{*'}Y^{*} + \mathbf{V}^{-1}\underline{\mu}_0)\}\right] \\
&\propto\ exp\left[-\frac{1}{2}\{\underline{\mu}'A\underline{\mu} - 2\underline{\mu}'(\sigma^{-2}Q^{*'}Y^{*} + \mathbf{V}^{-1}\underline{\mu}_0)\}\right] \\
&\propto\ N\left(\mathbf{A}^{-1}(\sigma^{-2}Q^{*'}Y^{*} + \mathbf{V}^{-1}\underline{\mu}_0),\ \mathbf{A}^{-1}\right)
\end{aligned}$$

The full conditional distributions of σ^2 (3.6)

$$\sigma^2|\cdot \;\sim\; IG\left(\frac{a+n}{2}, \frac{2}{b+(Y^*-Q^*\underline{\mu})'(Y^*-Q^*\underline{\mu})}\right)$$

Derivation >

$$\begin{aligned}
\sigma^2|\cdot \;&\propto\; (\frac{1}{\sigma^2})^{\frac{a}{2}+1} exp(-\frac{b}{2\sigma^2}) \times (\frac{1}{\sigma^2})^{\frac{n}{2}} exp\left[-\frac{1}{2\sigma^2}(Y^*-Q^*\underline{\mu})'(Y^*-Q^*\underline{\mu})\right] \\
&\propto\; (\frac{1}{\sigma^2})^{(\frac{a+n}{2}+1)} exp\left[-\frac{b+(Y^*-Q^*\underline{\mu})'(Y^*-Q^*\underline{\mu})}{2\sigma^2}\right] \\
&\propto\; IG\left(\frac{a+n}{2}, \frac{2}{b+(Y^*-Q^*\underline{\mu})'(Y^*-Q^*\underline{\mu})}\right)
\end{aligned}$$

The full conditional distribution of $\bar{\omega}$ (3.7)

$$\bar{\omega}|\cdot \sim IG\left(\frac{n_1+a_w}{2},\; \frac{2}{\sum_{i\in J}\frac{(y_i-\mu_0-\mu_1 s_i)^2}{\sigma^2}+b_w}\right) \times I(\bar{\omega}>1)$$

Derivation >

$$\begin{aligned}
\bar{\omega}|\cdot \;&\propto\; (\frac{1}{\bar{\omega}})^{\frac{a_w}{2}+1} exp(-\frac{b_w}{2\bar{\omega}}) \times (\frac{1}{\bar{\omega}\sigma^2})^{\frac{n_1}{2}} exp\left[-\frac{1}{2}\sum_{i\in J}\frac{(y_i-\mu_0-\mu_1 s_i)^2}{\sigma^2\bar{\omega}}\right] \\
&\propto\; (\frac{1}{\bar{\omega}})^{\frac{a_w+n_1}{2}+1} exp\left[-\frac{b_w+\sum_{i\in J}\frac{(y_i-\mu_0-\mu_1 s_i)^2}{\sigma^2}}{2\bar{\omega}}\right] \\
&\propto\; IG\left(\frac{n_1+a_w}{2},\; \frac{2}{\sum_{i\in J}\frac{(y_i-\mu_0-\mu_1 s_i)^2}{\sigma^2}+b_w}\right)
\end{aligned}$$

The full conditional distributions of $\eta = (p_{00}, p_{11})$ (3.8) and (3.9)

$$\begin{aligned} p_{00}|s &\sim beta(n_{00} + u_{00} - 1, n_{01} + u_{01} - 1) \\ p_{11}|s &\sim beta(n_{11} + u_{11} - 1, n_{10} + u_{10} - 1) \end{aligned}$$

Derivation >

$$\begin{aligned} p_{00}|S &\propto p_{00}^{n_{00}-1}(1-p_{00})^{n_{01}-1} p_{00}^{u_{00}-1}(1-p_{00})^{u_{01}-1} \\ &\propto p_{00}^{(n_{00}-u_{00}-1)-1}(1-p_{00})^{(n_{01}-u_{01}-1)-1} \\ &\propto beta(n_{00} + u_{00} - 1, n_{01} + u_{01} - 1) \\ p_{11}|S &\propto p_{11}^{n_{11}-1}(1-p_{11})^{n_{10}-1} p_{11}^{u_{11}-1}(1-p_{11})^{u_{10}-1} \\ &\propto p_{11}^{(n_{11}-u_{11}-1)-1}(1-p_{11})^{(n_{10}-u_{10}-1)-1} \\ &\propto beta(n_{11} + u_{11} - 1, n_{10} + u_{10} - 1) \end{aligned}$$

APPENDIX C

The purpose of this appendix is to provide the proof of (4.1) and derivation of full conditional distributions in Chapter IV.

The full conditional distributions of $\eta = (\lambda_0, \lambda_1)$ (4.8)

$$\lambda_j|\cdot \sim G(\bar{d}_j + a_j, (m_j + \frac{1}{b_j})^{-1}) \quad j \in \{0, 1\}$$

where, $\bar{d}_j$ is the number of y_i's whose state are j and m_j is the number of segments whose state are j at the previous iteration.

Derivation >

$$\begin{aligned} \lambda_j|\cdot &\propto \prod_{m_j} \left[\frac{\lambda_j^d e^{-\lambda_j}}{d!} \right] \times \lambda_j^{a_j - 1} e^{-\lambda_j / b_j} \\ &\propto \lambda_j^{\bar{d}_j + a_j - 1} e^{-\lambda_j (m_j + \frac{1}{b_j})} \\ &\propto G(\bar{d}_j + a_j, (m_j + \frac{1}{b_j})^{-1}) \end{aligned}$$

Proof of proposition 4.1

$$p_{ii}(d) = \begin{cases} 1 - P(d|s_{t-1} = i) & d = 1 \\ \frac{1-\sum_{k=1}^{d} P(k|s_{t-1}=i)}{1-\sum_{k=1}^{d-1} P(k|s_{t-1}=i)} & d > 1 \end{cases}$$

Proof >

It is straightforward to write

$$P(d|s_{t-1} = i) = \begin{cases} 1 - p_{ii}(d), & d = 1 \\ \prod_{k=1}^{d-1}(1 - p_{ii}(d))p_{ii}(k), & d > 1 \end{cases}$$

Since $P(d|s_{t-1} = i)$ is represented by the duration specific $p_{ii}(d)$ for each d, the probabilities $p_{ii}(d)$ can be expressed

$$\begin{aligned}
p_{ii}(1) &= 1 - P(1|s_{t-1} = i) \\
p_{ii}(2) &= 1 - \frac{P(2|s_{t-1} = i)}{p_{ii}(1)} = 1 - \frac{P(2|s_{t-1} = i)}{1 - P(1|s_{t-1} = i)} \\
&= \frac{1 - P(1|s_{t-1} = i) - P(2|s_{t-1} = i)}{1 - P(1|s_{t-1} = i)} \\
p_{ii}(3) &= 1 - \frac{P(3|s_{t-1} = i)}{p_{ii}(1)p_{ii}(2)} = 1 - \frac{P(3|s_{t-1} = i)}{1 - P(1|s_{t-1} = i) - P(2|s_{t-1} = i)} \\
&= \frac{1 - P(1|s_{t-1} = i) - P(2|s_{t-1} = i) - P(3|s_{t-1} = i)}{1 - P(1|s_{t-1} = i) - P(2|s_{t-1} = i)} \\
&\vdots \\
p_{ii}(d) &= 1 - \frac{P(d|s_{t-1} = i)}{\prod_{k=1}^{d-1} p_{ii}(k)} = 1 - \frac{P(d|s_{t-1} = i)}{1 - \sum_{l=1}^{d-1} P(l|s_{t-1} = i)} \\
&= \frac{1 - \sum_{k=1}^{d} P(k|s_{t-1} = i)}{1 - \sum_{l=1}^{d-1} P(l|s_{t-1} = i)} \\
&\quad \text{or} \\
p_{ii}(d) &= \frac{P(\text{duration} \quad \text{of} S_{t-1} = i > d)}{P(\text{duration} \quad \text{of} S_{t-1} = i > d - 1)}
\end{aligned}$$

VITA

Kyounghwa Bae, daughter of Chun-Ho Bae and Young-Ja Kim, was born on October 4, 1974, in PoHang, Korea. She graduated from PoHang girl's high school, Korea in 1993. She recieved a Bachelor of Science degree in statistics from Korea University in Seoul, Korea. She recieved a Master of Science in statistics from Korea University in Seoul, Korea under the direction of You-Sung Park in 1999. She continued her study in statistics under the direction of Professor Bani K. Mallick and Assistant professor Christine G. Elsik, and recieved a Doctor of Philosophy degree in statistics from Texas A&M University in May 2005.

Her permanent address is

135-14 Jegi2-Dong DongDaeMun-Gu

Seoul, Korea