Optimization Methods and Software
Vol. 00, No. 00, Month 200x, 1–15

Applications of Gibbs sampling in bioinformatics

Qizheng Sheng[†], Gert Thijs, Yves Moreau and Bart De Moor
Department of Electrical Engineering, Katholieke Universiteit Leuven
Kasteelpark Arenberg 10, 3001 Leuven-Heverlee, Belgium

(Received 00 Month 200x; In final form 00 Month 200x)

Gibbs sampling is a Markov chain Monte Carlo method for joint distribution estimation when the full conditional distributions of all the concerned random variables are available. The Gibbs sampling procedure iteratively draws samples from the full conditional distributions. The samples collected in this way are guaranteed to converge to the true joint distribution as long as there is no zero-probability in the target joint distribution.

Gibbs sampling strategy has been applied to Bayesian hierarchical models in bioinformatics. The first introduction of the methodology is its application to the motif-finding problem in DNA sequence analysis. We have recently applied this strategy to the analysis of gene expression data and have obtained biologically interpretable results.

This paper serves as a brief review for the applications of Gibbs sampling in the field of bioinformatics. We first discuss the working mechanism of Gibbs sampling. We then introduce some essential concepts needed for understanding the biological problems under concern. Finally, the models and the Gibbs sampling schemes for both the motif-finding problem and the biclustering problem of gene expression data are reviewed.

Index to information contained in this article

1 Introduction

Gibbs sampling is a technique to draw samples from a join distribution based on the full conditional distributions of all the associated random variables.

[†] To whom correspondences should be sent. Email: qizhengl.sheng@esat.kuleuven.ac.be.

The research is supported by Research Council KUL (GOA-Mefisto 666, GOA AMBioRICS, IDO, IOTA Oncology, Genetic networks, and PhD/postdoc and fellow grants); Flemish Government (FWO: PhD/postdoc grants, projects G.0115.01, G.0240.99, G.0407.02, G.0413.03, G.0388.03, G.0229.03, G.0241.04, G.0499.04, and research communities—ICCoS, ANMMM and MLDM; IWT: PhD Grants, STWW-Genprom, GBOU-McKnow, GBOU-SQUAD, and GBOU-ANA); Belgian Federal Science Policy Office(IUAP P5/22); and EU-RTD: (FP5-CAGE, ERNSI, FP6-NoE, FP6-IP, and e-Tumours).

Though the idea roots back to the work of Hasting (1970) [7], whose focus was on its Markov chain Monte Carlo (MCMC) nature, the Gibbs sampler was first formally introduced by Geman and Geman (1984) [6] to the field of image processing. The work caught the attention of the statistics society (especially boosted by the paper of Gelfand and Smith (1992) [4]). Since then, the applications of Gibbs sampling have covered both the Bayesian world and the world of classical statistics. In the former case, Gibbs sampling is often used to estimate posterior distributions, and in the latter, it is often applied to likelihood estimation [2].

In particular, Gibbs sampling has become a popular alternative to the expectation-maximization (EM) for solving the incomplete-data problem in the Bayesian context, where the associated random variables of interest include both the hidden variables (i.e., the missing data) and the parameters of the model that describe the complete data. To provide answers to this type of questions, EM is a numerical maximization procedure that climbs in the likelihood landscape aiming to find the model parameters and the hidden variables that maximize the likelihood function. In contrast, Gibbs sampling provides the means to estimate the target joint distribution of the hidden variables and the model parameters as a whole, and leave the estimation of the random variables for later (i.e. after the samples are drawn), where maximum a posterior (MAP) estimates are often used. Thus, Gibbs sampling sufferS less from the problem of local maxima than EM.

This property makes Gibbs sampling a suitable candidate for solving the model-based problems in bioinformatics, where the likelihood function usually consists of a large amount of modes due to the high complexity of the data. In this paper, we show the applications of Gibbs sampling to the hierarchical Bayesian models that address an important problem in systems biology. The goal is to discover regulation mechanism of genes. A typical framework by means of computational biology for this kind of study is composed of two steps. In the first step groups of genes that share similar expression profiles (which measured by the microarray technology) are found. (These genes are called to be coexpressed). This is done by performing clustering algorithms to the gene expression profiles (i.e., microarray data). The second step is based on the general assumption that coexpression implies coregulation. For each group of genes found in the first step, the DNA sequences that are related to the regulation of these genes are extracted, and common patterns of these sequences (called motifs) are seeked. The positions of these conserved motifs are likely to be the binding sites of transcription factors, which are the executors of the gene regulation mechanism. We show in this paper that the Gibbs sampling strategy can be applied to both the clustering of microarray data (particularly, the biclustering of microarray data, the idea of which is explained in more details in Section 5), and the motif finding problem of DNA sequences.

We will first review the working mechanism of Gibbs sampling. Then some basic biological concepts for understanding the biological problems of interest are introduced. Because Gibbs sampling has become the method-of-choice for the motif-finding problem in DNA sequence analysis, and our idea of applying Gibbs sampling to the biclustering of microarray data was inspired by this success, we will discuss the application of Gibbs sampling in the motif-finding problem first in Section 4 and then go into details about its applications in the biclustering problem of gene expression profiles in Section 5.

2 Gibbs sampling

Gibbs sampling allow statisticians to avoid the tedious and sometimes nontrivial mathematical calculations of integrals in obtaining the join distribution, by sampling directly from the full conditional distributions. (Because the same mechanism applies to both models for discrete data and models for continuous data, we use the terms "distribution" and "density" interchangeably). Suppose that we want to draw samples for the set of random variables $x_1, x_2, \ldots, x_K$, but that the marginal distributions (and thus the joint distribution) are too complex to directly sample from. Suppose also that the full conditional distributions $p(x_i \,|\, x_j;\, j \neq i)$ (for $i = 1, \ldots, K$) which can easily be sampled from, are available. Starting from initial values $x_1^{(0)}, x_2^{(0)}, \ldots, x_K^{(0)}$, the Gibbs sampler draws samples of the randome variables in the following manner,

$$\begin{aligned}
x_1^{(t+1)} &\sim p(x_1 \,|\, x_2 = x_2^{(t)}, \ldots, x_K = x_K^{(t)}) \\
x_2^{(t+1)} &\sim p(x_2 \,|\, x_1 = x_1^{(t+1)}, x_3 = x_3^{(t)}, \ldots, x_K = x_K^{(t)}) \\
\vdots\ & \ \vdots \\
x_i^{(t+1)} &\sim p(x_i \,|\, x_1 = x_1^{(t+1)}, \ldots, x_{i-1} = x_{i-1}^{(t+1)}, x_{i+1} = x_{i+1}^{(t)}, \ldots, x_K = x_K^{(t)}) \\
\vdots\ & \ \vdots \\
x_k^{(t+1)} &\sim p(x_K \,|\, x_1 = x_1^{(t+1)}, \ldots, x_{K-1} = x_{K-1}^{(t+1)}),
\end{aligned}$$

where t denotes the iterations.

Geman and Geman (1984) [6] shows that as $t \to \infty$, the distribution of $(x_1^{(t)}, \ldots, x_k^{(t)})$ converges to $p(x_1, \ldots, x_K)$. Equivalently, as $t \to \infty$, the distribution of $x_i^{(t)}$ converges to $p(x_i)$ (for $i = 1, \ldots, K$).

2.1 *The Markov chain property*

The convergence of samples drawn by the Gibbs sampler relies on the fact that these samples form Markov chains. In other words, $\Big((x_1^{(1)}, \ldots, x_k^{(1)}), \ldots, (x_1^{(t)}, \ldots, x_k^{(t)})\Big)$ as well as $(x_i^{(1)}, \ldots, x_i^{(t)})$ are Markov chains, where $(x_1^{(t)}, \ldots, x_k^{(t)})$ and $x_i^{(t)}$ are called the states of $(x_1, \ldots, x_k)$ and x_i respectively. The basic property of a Markov chain, take that of x_i for example, is

$$P(x_i^{(t+1)} \,|\, x_i^{(t)}, \ldots, x_i^{(0)}) = P(x_i^{(t+1)} \,|\, x_i^{(t)}), \tag{1}$$

which means that the future state of a random variable depends only on its current state but not on its past states. Writing

$$\begin{aligned} \pi_b(t+1) &= p(x_i^{(t+1)} = b) \\ \pi_a(t) &= p(x_i^{(t)} = a) \\ \text{and} \quad p(a \to b) &= p(x_i^{(t+1)} = b \,|\, x_i^{(t)} = a), \end{aligned}$$

we have

$$\pi_b(t+1) = p(a \to b)\pi_a(t). \tag{2}$$

$p(a \to b)$ is called the transition probability of going from state a to b (for random variable x_i). The probability transition matrix $\mathbf{P}$ is obtained by listing all the possible states for x_i along the rows and the columns, and fill the matrix with all the transition probabilities. (Note that this implies that each row of $\mathbf{P}$ sums to 1.) Thus to generalize Equation 2, we have

$$\pi(t+1) = \mathbf{P}\,\pi(t). \tag{3}$$

It is known that if the all the entries of $\mathbf{P}$ are above 0, an evolving Markov chain will reach a stationary distribution π^* after a sufficient amount of time [2], i.e.,

$$\pi^* = \mathbf{P}\,\pi^*. \tag{4}$$

Casella and George (1992) [2] gives a simple yet intuitive proof that the stationary distributions of the Markov chains generated by Gibbs sampling are the joint distribution $p(x_1, \ldots, x_k)$ and the marginal distributions $p(x_i)$, and that the probability transition matrices of these Markov chains can be derived from the full conditional distributions.

2.2 *The Monte Carlo property*

Only Those samples collected by the Gibbs sampler after the convergence is reached can be used for joint (or marginal) distribution estimation. The Gibbs sampling procedure performed before the convergence is reached is often referred to as the "burn-in procedure", and the procedure during which samples are collected will be called the "sampling procedure" hereafter. The samples collected in the sampling procedure enable us to calculate the expectation of a function $f(x_i)$ over the distribution $p(x_i)$. This is done by the Monte Carlo integration

$$\mathrm{E}_{p(x_i)}[f(x_i)] = \int f(x_i) \cdot p(x_i) \mathrm{d}x \approx \frac{1}{T} \sum_{t=1}^{T} f(x_i^{(t)}), \tag{5}$$

where t indexes the iterations in the sampling procedure, and T is the total number of samples collected. Thus, the expected value of x_i can be calculated as

$$\mathrm{E}_{p(x_i)}[x_i] = \int x_i \cdot p(x_i) \mathrm{d}x \approx \frac{1}{T} \sum_{t=1}^{T} x_i^{(t)}. \tag{6}$$

However, as illustrated by Gelfand and Smith (1990) [4] (using the Rao-Blackwell theorem), a more accurate estimate of the expected value of x_i is provided by

$$\mathrm{E}_{p(x_i)}[x_i] = \frac{1}{T} \sum_{t=1}^{T} \mathrm{E}_{p(x_i \,|\, x_1^{(t)}, \ldots, x_{i-1}^{(t)}, x_{i+1}^{(t)}, \ldots, x_k^{(t)})}[x_i]. \tag{7}$$

Similarly, the posterior distribution itself can be approximated by

$$\mathrm{E}[p(x_i)] = \frac{1}{T} \sum_{t=1}^{T} p(x_i \,|\, x_1^{(t)}, \ldots, x_{i-1}^{(t)}, x_{i+1}^{(t)}, \ldots, x_k^{(t)}). \tag{8}$$

With more generality, a better alternative for Equation 5 is

$$\mathrm{E}_{p(x_i)}[f(x_i)] = \frac{1}{T} \sum_{t=1}^{T} \mathrm{E}_{p(x_i \,|\, x_1^{(t)}, \ldots, x_{i-1}^{(t)}, x_{i+1}^{(t)}, \ldots, x_k^{(t)})}[f(x_i^{(t)})]. \tag{9}$$

The estimators obtained by Monte Carlo integration are unbiased estimators.

2.3 *Checking the convergence*

A key issue in using Gibbs sampling is to determine when the procedure has essentially converged. The number of iterations needed for the burn-in procedure varies from cases to cases. For a well mixed Markov chain whose samples cover most of the region of the random variable space, the convergence can be reached with in a few iterations. However, a bad starting point plus a multimodal target distribution with some of its probabilities close to zero can result in a poorly mixed chain so that only a small region of the random variable space is sampled for a long period of time. In this case, the number of burn-in iterations can easily reach a few thousand. In general, an optimal starting point close to the center of the marginal distribution can help in the accelerating the convergence. In addition, using multiple chains starting at independent positions of the random variable space can help to increase the coverage of the samples [5] and thus alleviate the problem of poorly mixed chains. Yet, convergence diagnostics are favorable in assisting the decision. Informal procedures of convergence diagnostics include inspecting the trace plot of the concerned variables or the evolution of likelihood. Various formal procedures has also been proposed. A good review on this issue is provided by Cowles and Carlin (1996) [3]. In the rest of this section, we will focus on one of the important issues for the convergence of a Markov chain—the autocorrelation.

One of the reasons that a Markov chain generated by the Gibbs sampler has a slow convergency is that the samples at successive iterations are not independent. This dependency implies that the variance (i.e., the accuracy) of the model obtained by averaging the parameters may be much higher than if the samples were independent. The autocorrelation time is the sum of the autocorrelation values for all positive lags and its square root gives the factor by which we must increase the number of iterates of the autocorrelated estimates to obtain the same accuracy as with independent estimates. Denoting by $\boldsymbol{\omega}^{(t)}$ the sets of parameters obtained at each iteration and by $\bar{\boldsymbol{\omega}} = (1/T)\sum_{t=1}^{T} \boldsymbol{\omega}^{(t)}$ the average set of parameters, the autocorrelation function for a lag of H can be estimated as

$$\hat{\rho}_m = \frac{\mathrm{Cov}(\boldsymbol{\omega}^{(t)}, \boldsymbol{\omega}^{(t+H)})}{\mathrm{Var}(\boldsymbol{\omega}^{(t)}} = \frac{\sum_{t=1}^{T-H}(\boldsymbol{\omega}^{(t)} - \bar{\boldsymbol{\omega}})(\boldsymbol{\omega}^{(t+H)} - \bar{\boldsymbol{\omega}})}{\sum_{t=1}^{T-H}(\boldsymbol{\omega}^{(t)} - \bar{\boldsymbol{\omega}})^2}. \tag{10}$$

In the frequent case where the autocorrelation function can be described as an autoregressive process, the autocorrelation time $\tau = \sum_{k=1}^{\infty} \hat{\rho}_k$ can be simplified to $\tau = (1+\hat{\rho}_1)/(1-\hat{\rho}_1)$. Such an estimate can be easily computed at the hand of the iterates of a run of the algorithm.

Another way to reduce the autocorrelation is to use the thinning of the Markov chain. Thinning with a factor J means that each J^{th} element in the chain will be used for the posterior summary statistics (see Equation 9). Another computational advantage of using the thinning procedure is that it saves the memory complexity of the Gibbs sampling procedure.

3 Some basic biology

DNA is known as the carrier of genetic information that is needed to conduct the synthesis of proteins—the workhorses in a living cell. Genes are fragments of the DNA sequence that carry such information. The first step of a protein synthesis procedure is the transcription of its corresponding gene, to a message RNA (mRNA). This step highly resembles the duplication of DNA molecules, which is made possible by the strict rule of base paring of the nucleotides, i.e., guanine ('G') can only be paired with cytosine ('C') and vice versa, while adenine ('A') can only be paired with thymine ('T'), and vice versa. In the case of transcription, RNA (instead of DNA) nucleotides are brought to be paired with the DNA templates also according to the rule of base pairing, the only difference is that uracil ('U') is paired with adenine (and vice versa), because there is no thymine in RNA. The second step is the translation of the mRNA to the protein. This step takes place in a ribosome where the mRNA is scanned three nucleotides (called a codon) at a time. Each possible combination of a codon (in total 64 possibilities) corresponds to one of the 20 amino acids. In this way, a peptide chain is assembled by the ribosome. The peptide chain is later folded into the resulting protein.

The above is only one part of the story that concerns the guidance of genes in the synthesis of proteins. The other part, however, is related to the regulative roles of proteins in the transcriptions of genes. The regulation of a gene is carried out by the transcription factors (TF, which are proteins themselves) that bind to the promoter region of the gene (which usually locates upstream, i.e. "in front", of a gene). These TFs can either enable or prohibit the binding of an RNA polymerase, which essentially opens the DNA double helix so that the transcription starts. (A good tutorial book for the beginners of biology is given by Lodish *e*t al. (2003) [9]).

One of the major research interests in bioinformatics is to untangle the gene regulation mechanism. The main-trend methodology for the task is based on the assumption that genes that share similar transcriptional behavior are under the same regulation program. (In other words, coexpression implies coregulation). Hence, the first step of the methodology is to identify genes that are coexpressed, and the second step is to look for the common transcription factor binding sites (TFBS) present in the promoter regions of these genes. The first step concerns the clustering of gene expression data—a matrix where the transcription levels of hundreds of thousands of genes under different experi-

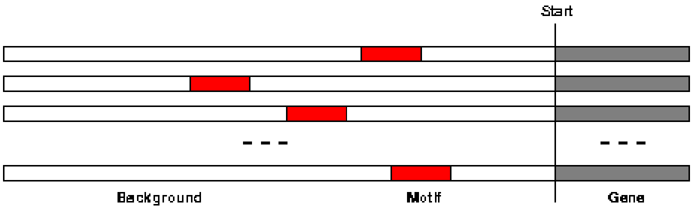

Figure 1. Conceptual plot of the motif-finding problem. The white regions containing the motifs are the promotor regions of the genes. The algorithm is only performed on the sequences of the promotor regions. These sequences are assumed to have only one common motif, and each sequence contains exactly one copy of the motif.

mental conditions are stored—measured by microarray technology. The second step concerns the analysis of the DNA sequence of the promotor regions of the coexpressed genes, where the particular patterns (i.e., motifs) of the sequence of the TFBSs as well as well their positions are to be revealed.

Probabilistic models are found to be suitable and useful [11, 15] for the analysis of both types of data. In the case of microarray data, the ability of probabilistic models to handel the high level of noise of microarrays in a principled way raises its popularity. In the case of motif-finding, probabilistic models not only are able to capture the fundamental identities of the motifs, but also provide the flexibility to allow subtle variations in the conserved motif sequences. Probabilistic models combined with Gibbs sampling for the parameterizations of the models and the missing data estimation have become the method-of-choice for motif-finding [15]. The efficiency of applying Gibbs sampling to the probabilistic models on microarray data has also been demonstrated recently [12, 13].

4 Gibbs sampling for motif-finding

To begin with, we suppose that for a given set of genes for which we assume to share the same regulation mechanism, the sequences of their promotor regions are available. We also suppose that there is only one motif (thus one TFBS) in common for these sequences, and that there is exactly one copy of the motif in each sequence, (see Figure 1 for an illustration).

To describe the problem probabilistically, we assume that the sequences $\mathcal{S} = \{\boldsymbol{S}_k \,|\, k = 1 \ldots N\}$ (where N is the number of sequences) are generated by a general background model except for the subsequences where the motif occurs, which is generated by a motif model. The background model is represented by a multinomial distribution,

$$\boldsymbol{\theta}_0 = [\theta_{\mathrm{A}0},\, \theta_{\mathrm{C}0},\, \theta_{\mathrm{G}0},\, \theta_{\mathrm{T}0}]^{\mathrm{T}}, \tag{11}$$

whose four entries provide the probability that a base of the sequence is generated by 'A', 'C', 'G' and 'T' respectively. We assume that the length of the motif is known to be W. The model of the motif is provided by a product multinomial distribution,

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_W], \tag{12}$$

where

$$\boldsymbol{\theta}_j = [\theta_{\mathrm{A},j}, \theta_{\mathrm{C},j}, \theta_{\mathrm{G},j}, \theta_{\mathrm{T},j}]^{\mathrm{T}}, \quad j = 1 \ldots W \tag{13}$$

is the parameter vector of the multinomial distribution for the j^{th} position in the motif.

Further, we use random variables $\mathcal{A} = \{a_k \,|\, k = 1 \ldots N\}$ for the starting positions of the motif in the sequences. These starting positions are the "missing data" of this problem. They are called the alignments of the motif. Note that for each of the sequences $\boldsymbol{S}_k$

$$\sum_{i=1}^{L_k} P(a_k = i) = 1, \tag{14}$$

where L_k is the length of $\boldsymbol{S}_k$.

Given the data, we are interested in finding the alignments of the motif as well as the model of the motif and the model of the background. Therefore, our target joint distribution is $P(\boldsymbol{\Theta}, \boldsymbol{\theta}_0, \mathcal{A} \,|\, \mathcal{S})$. Using Gibbs sampling to estimate this joint distribution means that we need to sample from the full conditional distributions of $\boldsymbol{\Theta}$, $\boldsymbol{\theta}_0$ and each a_k. However, when conjugate priors are applied (which is often a sensible choice), the full conditional distribution of $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}_0$ are in the form of Dirichlet distributions. Sampling from a Dirichlet distribution is not a trivial procedure and consumes a non-negligible amount of computation. Liu, Neuwald and Lawrence (1995) [8] demonstrated that $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}_0$ can be integrated out of the target distribution, and that the motif model and the background model can be obtained as a byproduct of the Gibbs sampling procedure for estimating $P(\mathcal{A} \,|\, \mathcal{S})$.

As shown in Liu, Neuwald and Lawrence (1995) [8], the final obtained full conditional distribution for a_k is

$$P(a_k = i \,|\, \mathcal{A}_{\bar{k}}, \mathcal{S}) = \frac{1}{Z} \prod_{j=1}^{w} \left(\frac{\hat{\boldsymbol{\theta}}_j}{\hat{\boldsymbol{\theta}}_0} \right)^{\mathrm{s}_{i+j-1}} . \tag{15}$$

In the above equation, $\mathcal{A}_{\bar{k}}$ denotes the alignments in all the sequences other than $\boldsymbol{S}_k$; Z is a term that ensures the validity of Equation 14; s_x corresponds to the x^{th} position in $\boldsymbol{S}_x$, and it is an index vector of length four with 1 on the entry corresponding to the nucleotide at position k and 0 for the rest of the entries; and $\hat{\boldsymbol{\theta}}_j$ and $\hat{\boldsymbol{\theta}}_0$ are respectively the vectors of posterior frequencies for observing the four nucleotides at the corresponding position of the current bicluster and at the current background. By "current", we mean that $\hat{\boldsymbol{\theta}}_j$ and $\hat{\boldsymbol{\theta}}_0$ are calculated using all the sequences other than $\boldsymbol{S}_k$. In addition, the power as well as the division of two vectors are carried out entry-wise in Equation 15. The equation shows that the full conditional probability for a certain position in the promotor sequence to be the alignment position is the proportional to the likelihood ratio between the case when its succeeding w positions are generated by the motif model and the case when these positions are generated by the background model.

The discussed method finds one motif at a time. In order to discover multiple motifs in a set of promotor sequences, found motifs are masked (i.e., the alignment positions of the found motif are not taken into consideration for further analysis) and Gibbs sampling is performed on the rest of the data.

In addition, the simple model discussed above can be modified to allow none or multiple copies of a motif in each sequence [14]. Furthermore, higher order hidden Markov chain (HMM) models are found to be useful in modelling the background sequences in order to produce more reliable results [14]. Combining these two features, this extended model as been implemented as the *MotifSampler*. Detailed examples and results concerning the performance and the efficiency of the tool can be found in Thijs (2003) [15].

5 Gibbs sampling for biclustering gene expression data

Microarray technology provides biologists the tool to take a snapshot of the transcriptional behavior of thousands of genes simultaneously. By using several microarrays and performing the experiments in different conditions, biologists are able to monitor the transcriptional behavior of the genes. Genes that behave similarly under different conditions (i.e. coexpressed genes) imply that they might share the same regulation mechanism, and further imply that they might share the same functions in cell.

The expression values measured by microarray experiments are usually put in a matrix, whose rows represent the genes and whose columns represent the conditions. To find genes that behave similarly across the experiments is thus equivalent to perform a clustering analysis of the rows. However, when the experiments compose a heterogeneous compendium, genes that share the same function do not necessarily have to behave similarly over all the conditions. Rather, their behavior would be close to each other under a subset of conditions

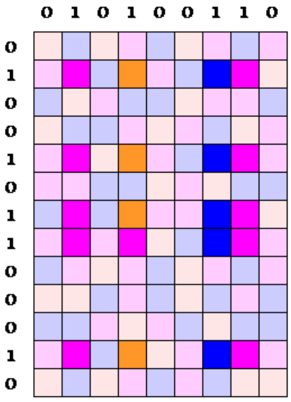

Figure 2. Conceptual plot of the biclustering problem: colors in the matrix represent different values. The data of the bicluster is highlighted. Binary labels are used to indicate if the row or the column belongs to the bicluster.

(which are relevant to the cell function) while their expression profiles could be totally noisy under the others. In this case, a biclusterng algorithm which groups genes based on only a subset of experimental conditions and in the meantime identifies the relevant conditions is more favorable. This biclustering problem also exists for the other dimension of the microarray matrix, i.e. to group experiments (e.g., patients) under each of which a sub set of genes have almost the same expression values. To differ between the two types of biclustering problems, we refer to the former as the biclustering of genes, and the latter as the biclusteirng of experiments.

To generalize the two problems, we transpose the microarray matrix in the case of biclustering experiments, so that the biclustering problem is to find set of rows in a matrix, whose data entries under each selected column (for the bicluster) are similar (see Figure 2 for an illustration).

The idea of introducing the Gibbs sampling strategy to the biclustering problem [13] was inspired by the success of Gibbs sampling in the motif-finding problem.

Putting the biclustering problem in the probabilistic framework, we use binary random variables to describe the association of the rows and columns to the bicluster—i.e., a “1” indicates that the row (or the column) belongs to the bicluster and a “0” for otherwise. The binary labels are the missing data in the problem. They are divided into two sets, $\{\mathcal{R} \,|\, r_i,\ i = 1 \ldots N\}$ for the rows (where N is the number of rows), and $\{\mathcal{C} \,|\, c_j,\ j = 1 \ldots M\}$ for the columns (where M is the number of columns).

The joint distribution of interest is

$$p(\mathcal{R}, \mathcal{C}, \mathcal{M}^{\mathrm{bcl}}, \mathcal{M}^{\mathrm{bgd}}, \,|\, \mathbf{D}), \tag{16}$$

where $\mathcal{M}^{\mathrm{bcl}}$ and $\mathcal{M}^{\mathrm{bgd}}$ denote respectively the model of the bicluster and the model of the background, and $\mathbf{D}$ denotes the data. Therefore, to carry out the Gibbs sampling strategy, full conditional distributions of the labels and the model parameters need to be derived. The derivation of these conditional distributions requires not only the specification of the data models (where the data refer to both the observed data, but also the missing data—the labels), but also their prior distribution. Conjugate priors are used for each concerned parameters.

The natural choice for the distribution of the labels are Bernoulli distributions, whose conjugate priors are Beta distributions—$Beta(\xi_{r0},\, \xi_{r1})$ for the row labels, and $Beta(\xi_{r0},\, \xi_{r1})$ for the column labels. The Bernoulli parameters of the model can be integrated out of the target distribution to simplify the calculation (which is actually already implied in Equation 16). The posterior distribution of a row label is

$$\begin{aligned}
&p(r_i = 1 \,|\, \mathcal{R}_{\bar{i}},\, \mathcal{C},\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}}, \mathbf{D})\\
&= \frac{p(\mathbf{D},\, r_i = 1,\, \mathcal{R}_{\bar{i}},\, \mathcal{C} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}})}{p(\mathbf{D},\, r_i = 1,\, \mathcal{R}_{\bar{i}},\, \mathcal{C} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}}) + p(\mathbf{D},\, r_i = 0,\, \mathcal{R}_{\bar{i}},\, \mathcal{C} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}})}\\
&= \frac{\gamma_i^r}{1 + \gamma_i^r}
\end{aligned}$$

where γ_i^r is the likelihood ratio between the case that the i^{th} row is generated by $\mathcal{M}^{\mathrm{bcl}}$ and the case when it is generated by $\mathcal{M}^{\mathrm{bgd}}$,

$$\gamma_i^r = \frac{p(\mathbf{D},\, r_i = 1,\, \mathcal{R}_{\bar{i}},\, \mathcal{C} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}})}{p(\mathbf{D},\, r_i = 0,\, \mathcal{R}_{\bar{i}},\, \mathcal{C} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}})} \quad i = 1 \ldots N. \tag{17}$$

In the above two equations, $\mathcal{R}_{\bar{i}}$ denotes the set of row labels except for the i^{th} row. Similarly, the conditional distribution of a column label is also in the form of likelihood ratio, where

$$\gamma_j^c = \frac{p(\mathbf{D},\, \mathcal{R},\, c_j = 1,\, \mathcal{C}_{\bar{j}} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}})}{p(\mathbf{D},\, \mathcal{R},\, c_j = 0,\, \mathcal{C}_{\bar{j}} \,|\, \mathcal{M}^{\mathrm{bcl}},\, \mathcal{M}^{\mathrm{bgd}})} \quad j = 1 \ldots M. \tag{18}$$

The first attempt of applying Gibbs sampling to the biclustering problem is to bicluster patients on discretized microarray data [13]. The data is discretized using a maximum-entropy principle, (i.e., the equal-frequency principle). For example, when discretizing microarray data into three bins (i.e., three discretized expression levels that may interpreted as being “high”, “medium” and “low”), for each gene profile, we assign the experiments with the lowest $\frac{1}{3}$ of the expression values to the first bin (corresponding to “low”), similarly, the $\frac{1}{3}$ experiments with the highest expression values to the third bin (corresponding to “high”), and finally, the rest of the experiments to the second bin (corresponding to “medium”). Besides the convenience of borrowing the models from the motif-finding problem, the reason for using of discretized data for the biclustering problem of patients is two-fold. First, comparing with the huge gene dimension (which is usually measured in thousands), microarray data usually contains much fewer experiments (whose number is usually no larger than a couple of hundreds). Using a normal distribution to a gene expression profile would often be found sensitive to outliers [10]. The equal-frequency principle discretization avoids the problem of outliers by significantly reducing the noise level in the data while reserving the most essential information for biologists. Second, the equal-frequency discretization takes care of the normalization of the experiments automatically, and hence provides the base for comparison between the experiments.

In this case, the model of the bicluster $\mathcal{M}^{\text{bcl}}$ is a product multinomial distribution,

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_W], \tag{19}$$

where W now is the number columns in the bicluster, and $\boldsymbol{\theta}_j = [\theta_{1,j}, \theta_{2,j}, \ldots, \theta_{Q,j}]^{\text{T}}$ (for $j = 1 \ldots W$, and Q is the number of bins used for discretization). The equal-frequency discretization justifies the use of a universal multinomial background model $\mathcal{M}^{\text{bgd}}$ for data under all the experimental conditions

$$\boldsymbol{\theta}_0 = [\theta_{1,0}, \theta_{2,0}, \ldots, \theta_{Q,0}]^{\text{T}}. \tag{20}$$

As explained in Section 4, the parameters in $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}_0$ follow Dirichlet distributions which are difficult to sample from. Therefore, $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}_0$ are integrated out of the target joint distribution.

It can be shown [13] that the likelihood ratio for calculating the full conditional distribution of a row label is

$$\gamma_i^r = \prod_{j \in \mathcal{C}',\, \mathcal{C}' = \{c_k = 1 | c_k \in \mathcal{C}\}} \left(\frac{\hat{\boldsymbol{\theta}}_j}{\hat{\boldsymbol{\theta}}_0} \right)^{\delta_{i,j}} \cdot \frac{V_{\bar{i}} + \xi_{r1}}{N - 1 - V_{\bar{i}} + \xi_{r0}} \quad i = 1 \ldots N. \tag{21}$$

The first term in the above equation resembles that of Equation 15, where the notations have similar explanations. $\hat{\boldsymbol{\theta}}_j$ and $\hat{\boldsymbol{\theta}}_0$ are now the posterior frequencies of the current bicluster and the background (i.e. evaluated on the rest of the data other than those in the i^{th} row). $\delta_{i,j}$ is an index vector of length

Q, whose entries are 0 except for the entry corresponding to the value of the $\{i, j\}^{\text{th}}$ data point in the matrix. The second term of Equation 21 comes from the integration of the Bernoulli distribution for $p(r_i)$, where $V_{\bar{i}}$ denotes the number of rows in the current bicluster.

The likelihood ratio for calculating a column label is in a more complicated form

$$\gamma_j^c = \frac{f(\mathbf{D}[\mathbf{r}, j]) \cdot f(\mathbf{D}[\bar{\mathbf{r}}, j])}{f(\mathbf{D}[\cdot, j])} \cdot \frac{W_{\bar{j}} + \xi_{c1}}{M - 1 - W_{\bar{j}} + \xi_{c0}} \quad j = 1 \ldots M, \tag{22}$$

where $\mathbf{r} = \{k \,|\, r_k = 1 \wedge r_k \in \mathcal{R}\}$ and $\bar{\mathbf{r}} = \{k \,|\, r_k = 0 \wedge r_k \in \mathcal{R}\}$, $\mathbf{D}[\mathbf{u}, \mathbf{v}]$ denotes the data at rows $\mathbf{u}$ and columns $\mathbf{v}$ ($\mathbf{u}$ and $\mathbf{v}$ are vectors of indices), $f(\cdot)$ is a function that involves the evaluation of gamma functions on the specified data, and $W_{\bar{i}}$ denotes the number of columns in the current bicluster. See Sheng, Moreau and De Moor (2003) [13] for more details on the full conditional distributions, and also for some examples on applying the model to discrete microarray data.

When applying the Gibbs sampling strategy to the biclustering of genes, Gaussian likelihood with Gaussian-Wishart prior can be used to model both the bicluster and the background [12]. This choice is based on not only the analytical convenience of normal models, but also the consensus that the assumption for fitting a normal distribution to the gene expression measurements in a given situation is considered to be reasonable especially when a proper preprocessing procedure has been applied to the microarray data [1]. In this case, The full conditional distributions of the binary labels are related to the Gaussian likelihood ratios, and the conditional distributions of the model parameters are in the form of Gaussian-Wishart distributions which can be easily sampled from. Because of the limitation on the length of the paper, we refer to Sheng *et al.* (2005) [12] for more detail in this regard.

Multiple biclusters can be found by masking the found biclusters as well. For both of the biclusering problems, we choose to mask the experimental conditions by skipping to sample the experiment labels of a found bicluster, which are permanently set to 0. The reason for masking the experiments instead of masking the genes is that a genes might have multiple functions.

6 Conclusion

The general methodology that we discussed in Section 2 can be applied directly to the full conditional distributions for solving the motif-finding problem (Section 4) and the biclustering problem (Section 5). Of course, other models (such as t-distributions for the biclustering of experiments) can be explored for tailoring the models to fit the nature of biological data. Yet, Gibbs sampling as a general methodology is well suitable for solving the incomplete data problems in bioinformatics.

EM is another alternative for solving this type of problem. However, we favor the Gibbs sampling approach for the applications in bioinformatics because Gibbs sampling is more suitable to deal with the vast amount of local modes in the models due to the highly complex nature of biological data. Instead of climbing in the likelihood landscape (which is the case for EM), Gibbs sampling pictures the posterior distribution of the concerned random variables (i.e., both the hidden variables and the model parameters) as a whole, and MAP estimation is made by means of Monte Carlo integration. In this way, Gibbs sampling highly reduces the chance to find local maxima comparing with EM. In addition, taking into account the fact that it is hard to estimate how many multiple runs are needed for EM to find the global maximum solution, we consider Gibbs sampling to be practically more efficient for applications in bioinformatics.

References

[1] Baldi P. and Long A. D., 2001, A Bayesian framework for the analysis of microarray expression data: regularized *t*-test and statistical inference of gene changes. *Bioinformatics*, **17**, 509–519.

[2] Casella G. and George E. I., 1992, Explaining the Gibbs Sampler. *Journal of the American Statistical Association,* **46**, 167–174.

[3] Cowles M. K. and Carlin B., 1996, Markov chain Monte Carlo convergence diagnositcs: a comparaitve review. *Journal of the American Statistical Association*, **91**, 883–904.

[4] Gelfand A. E. and Smith F. M., 1990, Sampling-based approaches to calculating marginal densities. *Journal of the America Statistical Association*, **85**, 398–409.

[5] Gelman A. and Rubin D. B., 1992, Inference from iterative simulation using multiple sequences. *Statistical Science*, **7**, 457–472.

[6] Geman S. and Geman D., 1984, Stochastic relaxation, Gibbs distribution, and the Bayes restoration of images. *IEEE transactions on pattern analysis and machine intelligence*, **6**, 721–741.

[7] Hastings, W. K., 1970, Monte Carlo sampling methods using Markov chains and their applications, *Biometrika*, bfseries 87, 97-109.

[8] Liu L. S., Neuwald A. F. and Lawrence C. E., 1995, Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. *Journal of the American Statisticians*, **90**, 1156–1170.

[9] Lodish H., Berk A., Matsudaira P., Kaiser C. A., Krieger M., Scott M. P., Zipursky L. and Darnell J., 2003, Molecular Cell Biology (Fifth edition). WH Freeman publishers.

[10] McLachlan G. J., Bean R. W. and Peel D., 2002, A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, **18**(3), 413–422.

[11] Segal E., Taskar B., Gasch A., Friedman N., and Koller D., 2001, Rich probabilistic models for gene expression, *Bioinformatics*, **17**(Suppl. 1), S243–S252.

[12] Sheng Q., Lemmens K., Marchal K., De Moor B. and Moreau Y., 2005, Query-driven biclustering of microarray data by Gibbs sampling. Internal report 05-33, Department of Electrical Engineering (ESAT-SCD-SISTA), Katholieke Universiteit Leuven, Belgium

[13] Sheng Q., Moreau Y. and De Moor B., 2003, Biclustering microarray data by Gibbs sampling. *Bioinformatics*, **19**(Suppl. 2), II196–II205.

[14] Thijs G., Marchal K., Lescot M., Rombauts S., De Moor B., Rouzé B. and Moreau Y., 2002, A Gibbs sampling method to detect overrepresented motifs in the upstream regions of coexpressed genes. *Journal of Computational Biology*, **9**, 447–464.

[15] Thijs G., 2003, Probabilistic methods to search for regulatory elements in sets of coregulated genes. PhD thesis, Katholieke Universiteit Leuven, Belgium.