Probabilistic Graphical Models 10-708
Homework 4: Due April 14, 2014 at 4 pm

Directions. This homework assignment covers the material presented in Lectures 13-18. You must complete all four problems to obtain full credit. To submit your assignment, please upload a pdf file containing your writeup and a zip file containing your code to Canvas by 4 pm on Monday, April 14th. We highly encourage that you type your homework using the LaTeX template provided on the course website, but you may also write it by hand and then scan it.

Important Note. You homework writeup should be saved as a pdf file. Before submitting your assignment, please double check that everything you want to include shows up in the compiled pdf. You must also make sure that this document is legible. This means that if you don't use LaTeX, your handwriting must be neat and your answers must be organized. Next, place all of your code in a directory and compress it into a zip file. Do not place tex files or anything else inside of this directory. Finally, when submitting the assignment, please separately attach both the pdf file and the zip file to your Canvas submission. This makes it much easier for us to grade the non-programming questions because your writeup will load on the page in the grading tool that we use. If you do not follow these instructions, e.g. if you place your writeup inside of the zip file or write illegibly, we will take 5 points off of your final homework grade, and possibly more if we cannot read your answers to certain questions.

1 Gibbs Sampling [25 points]

Part 1: Stationary Distribution [10 points]

A distribution π is a stationary distribution for a Markov chain with transition kernel $\mathcal{T}$ if it satisfies the following condition:

$$\pi(\boldsymbol{x}') = \sum_{\boldsymbol{x}} \pi(\boldsymbol{x})\, \mathcal{T}(\boldsymbol{x} \to \boldsymbol{x}')$$

Using the above definition, show that the posterior distribution for a graphical model with hidden variables $\boldsymbol{X}$ and evidence $\boldsymbol{E} = \boldsymbol{e}$ is a stationary distribution of the Markov chain that is induced by running Gibbs sampling on that model.

Hint: You can do this by simply proving that the posterior is a stationary distribution of the Markov chain induced by the local transition kernel for each individual variable, denoted $\mathcal{T}_i\left(x_i \to x_i'\right)$. Since Gibbs sampling constructs a global Markov chain by combining multiple such kernels, we can prove that the posterior is a stationary distribution for the multiple-kernel chain by showing that it is a stationary distribution for each of the individual kernels. In this problem, you don't need to prove the last part; just show that the posterior is a stationary distribution for the Markov chain defined by each local kernel.

Part 2: Special Case of Metropolis-Hastings [10 points]

Show that Gibbs sampling is a special case of the Metropolis-Hastings algorithm. Give the exact proposal distribution and acceptance probability used in Gibbs sampling, and show that when these are used in a Metropolis-Hastings procedure, they yield the same transition kernel as the one used in Gibbs sampling.

Hint: Once again, it is sufficient to provide the proposal distribution and acceptance probability for each local transition kernel, $\mathcal{T}_i$.

Part 3: Variants of Gibbs Sampling [5 points]

What is the difference between regular Gibbs, block Gibbs, and collapsed Gibbs sampling? Provide a brief description of the block and collapsed Gibbs algorithms, and make sure the difference between each of these and regular Gibbs sampling is clear.

2 Collapsed Gibbs Sampling for LDA [25 points]

In this problem, we will derive collapsed Gibbs sampling equations for Latent Dirichlet Allocation (LDA) with conditional probabilities:

$$\boldsymbol{\phi}_k \sim \text{Dirichlet}(\boldsymbol{\beta}) \tag{1}$$
$$\boldsymbol{\theta}_i \sim \text{Dirichlet}(\boldsymbol{\alpha}) \tag{2}$$
$$z_{ji}|\theta_i \sim \text{Discrete}(\boldsymbol{\theta}_i) \tag{3}$$
$$d_{ji}|z_{ji}, \boldsymbol{\phi}_{z_{ji}} \sim \text{Discrete}(\boldsymbol{\phi}_{z_{ji}}) \tag{4}$$

Here j is the index for words ($\boldsymbol{d}_i = \{d_{1i}, \ldots, d_{Ni}\}$), i is the index for documents, and k is the index for topics. Also, we use the following notation: $N_{wki} = |\{j : d_{ji} = w, z_{ji} = k\}|$ (total number of times the word w in document i is assigned to the topic k), $N_{ki} = \sum_w N_{wki}$, and $N_{wk} = \sum_i N_{wki}$. We use superscript $(-ji)$ (e.g. $N_{wki}^{(-ji)}$) to indicate that the corresponding word d_{ji} in document i is not counted in N_{wki}.

1. [3 pts] Write down $P(\boldsymbol{d}|\boldsymbol{z}, \boldsymbol{\beta})$ and $P(\boldsymbol{z}|\boldsymbol{\alpha})$ using their conditional probabilities. (Hint: Integrate out $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$, respectively)

2. [2 pts] Exact probabilistic inference on $p(\boldsymbol{z}|\boldsymbol{d})$ is infeasible. Explain the reason why the exact inference is infeasible.

3. [10 pts] Since exact inference is infeasible, we will use approximate inference. In particular, in this problem, we are interested in collapsed Gibb's sampling (It is called "collapsed" Gibb's sampling since $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ are integrated out in the inference procedure). Prove the following LDA collapsed Gibb's sampling equation:

$$p(z_{ji} = k|\boldsymbol{z}\backslash z_{ji}, \boldsymbol{d}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto (N_{ki}^{(-ji)} + \alpha_k)\frac{N_{wk}^{(-ji)} + \beta_w}{N_k^{(-ji)} + \sum_w \beta_w},$$

   where $w = d_{ji}$.
   (Hint: $\Gamma(x+1) = x\Gamma(x)$)

4. [5 pts] Note that $\boldsymbol{\theta}_i$ (document-topic proportion) and $\boldsymbol{\phi}_k$ (topic-word distribution) can be represented by using only z_{ji} (topic assignment for each word d_{ji} in document i). Let $\tilde{z} \in \{1, ..., K\}$ be a new topic assignment drawn from $p(\tilde{z}|\{z_{ji}\}_{j=1}^{N_i}, \boldsymbol{\alpha})$. Write down $\theta_{ik} := p(\tilde{z} = k|\{z_{ji}\}_{j=1}^{N_i}, \boldsymbol{\alpha})$. Similarly, let $\tilde{w}$ be a token drawn from $p(\tilde{w}|\tilde{z}, \{z_{ji}, w_{ji}\}_{j=1}^{N_i}, \boldsymbol{\beta})$, write

down $\phi_{kw} := p(\tilde{w} = w|\tilde{z} = k, \{z_{ji}\}_{j=1}^{N_i}, \boldsymbol{\beta})$ where w indexes the vocabulary. Together, θ_{ik} and ϕ_{kw} fully specify the generative process described earlier. You don't need to show the derivation, but you are welcome check out the Wikipedia page on Dirichlet-multinomial distribution and give it a shot.

5. [5 pts] Write down pseudo-code for LDA collapsed Gibbs Sampling.

3 Gibbs Sampling vs Metropolis-Hastings

In this question, you will compare the performance of Gibbs sampling and Metropolis-Hastings with a real programming task.

Consider N i.i.d data points $x^N = \{x_1, \ldots, x_N\}$ drawn from a two component Gaussian mixture model of the following form:

$$x^N \sim \frac{1}{2}\mathcal{N}(\mu_1, 1) + \frac{1}{2}\mathcal{N}(\mu_2, 1). \tag{5}$$

3.1 Part 1: An Improper Prior

Consider the (improper) prior on μ_1 and μ_2, written:

$$p(\mu_1, \mu_2) \propto 1. \tag{6}$$

Show that this prior is improper, i.e. show that the posterior does not have a finite integral.

3.2 Part 2: Deriving Gibbs Sampling

Next, consider the prior on μ_1 and μ_2, written:

$$p(\mu_1, \mu_2) = \mathcal{N}(\mu_1|0, \tau)\mathcal{N}(\mu_2|0, \tau) \tag{7}$$

where τ is a fixed constant. Write down a Gibbs sampling algorithm for this problem (i.e. write down the conditional distributions from which you must sample in the Gibbs sampling algorithm).

3.3 Part 3: Running Gibbs Sampling

Generate 100 observations in the following manner: set $\mu_1 = -5$, $\mu_2 = 5$, and $\tau = 10$, and perform forward sampling to generate the observations.

Next, use your Gibb sampling algorithm from Part 2 to generate samples from the posteriors $p(\mu_1|x^N)$ and $p(\mu_2|x^N)$. For each posterior, generate 2000 samples, and then discard the first 500 as burn-in. Show kernel-density estimates of the resulting distributions.

3.4 Part 4: Running Metropolis-Hastings

Next, you will sample from the same model with a Metropolis-Hastings algorithm. First, choose a symmetric proposal distribution for the MH algorithm (and write this down). Next, generate samples from each of the two posteriors as before. Same as before, generate 2000 samples, and then discard the first 500 as burn-in. Show kernel-density estimates of the resulting distributions.

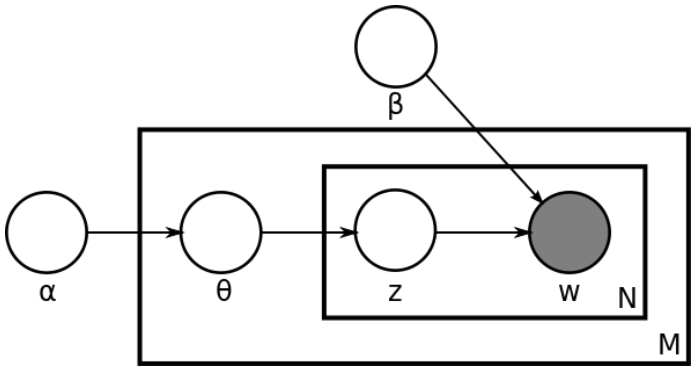

Figure 1: The LDA graphical model

4 Variational Inference in Latent Dirichlet Allocation (LDA) [25 points]

The LDA graphical model (Figure 1) was discussed in class. The most popular use for LDA is in modeling a document collection by topics, however, LDA-like models can also be used for various other modeling tasks. In this question, we will apply LDA to the problem of discovering human ancestry.

In applications of population genetics, it is often useful to classify individuals in a sample into populations. An underlying assumption is that there are K ancestor populations, and each individual is an admixture of the ancestor populations. For example, in studies of human evolution, the population is often considered to be the unit of interest, and a great deal of work has focused on learning about the evolutionary relationships of modern populations.

For each individual, we measure some genetic data about them, called genotype data. Each genotype is a locus that can take a discrete count value, individuals with similar genotypes are expected to belong to the same ancestor populations. We can derive the admixture coefficients (θ) for each individual by running an LDA model, where the documents are the individuals, and the words are the genotype.

In this question, we will implement variational inference to infer the population mixture (θ) and the genotype ancestry (topic) assignments (z) for any individual. The variational distribution used to approximate the posterior (for a given individual i) is $q(\theta, z|\gamma, \phi) = q(\theta|\gamma)\prod_{n=1}^{N_i} q(z_n|\phi_n)$, where the Dirichlet parameter γ and the multinomial parameters $(\phi_1, \cdots, \phi_{N_i})$ are the free variational parameters (N_i is the number of non-zero genotype loci for this individual). See Figure 2 for a graphical representation.

The data matrix in **data.mat** provides data about $M = 100$ individuals, each represented by a vocabulary of $N = 200$ genotype loci. This data has been preprocessed into a count matrix D of size $M \times N$. D_{ij} is the number of occurrences of genotype j in individual i, and $\sum_j D_{ij}$ is the number of genotype loci in an individual.

We learnt the LDA topic model over $K = 4$ ancestor populations, and the inferred β matrix of size $N \times K$ has been stored in **beta_matrix** in **data.mat**. The value of α is 0.1.

In the writeup, report the following:

1. Report the variational inference update equations for estimating γ and ϕ (you don't have to derive them).
2. For individual 1, run LDA inference to find ϕ for each genotype locus, store it as a matrix of size $n_1 \times K$ (where $n_1 := \sum_{1j} I(D_{1j} \neq 0)$, $I(\cdot)$ being the indicator function, is the number

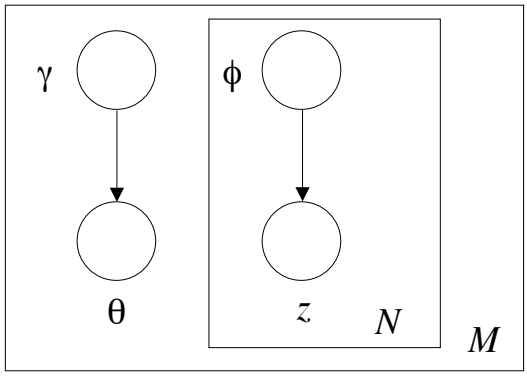

Figure 2: Graphical model representation of the variational distribution used to approximate the posterior in LDA.

of non-zero genotypes present in individual 1), and plot it as an image in your writeup (Use $imagesc(\phi); colorbar$ in matlab). Don't forget to show the colormap using the *colorbar* function to allow the colors in the image to be mapped to numbers!

3. We will construct a matrix Θ of size $M \times K$ to represent the ancestor assignments for all individuals in the population. For each individual i, run LDA inference to find γ, and store it as row of Θ, i.e. $\Theta_i = \gamma$. Visualize Θ as an image (Use $imagesc(\Theta); colorbar$ in matlab).
4. Report the number of iterations needed to get to convergence for running inference on all M individuals (check the convergence criteria in the "implementation hints" section below).
5. Report the time taken to run inference on all M individuals.
6. Repeat the experiment for $\alpha = 0.01$, $\alpha = 1$, $\alpha = 10$, and for each value of α, visualize the Θ matrix summarizing the ancestor population assignments for all individuals. Discuss the changes in the ancestor population assignments to the individuals as α changes. Does the mean number of iterations required for convergence for inference change as α changes?

Implementation hints:

1. If you use matlab, **beta** is a pre-defined function for the beta function, hence you might want to not use beta as a variable name to avoid overloading.
2. In this assignment, regular updates will most likely work fine, since the vocabulary size (number of genotype loci) is so small, but if you wanted a usable implementation for other problems, updating probabilities would need to be done in log-space to avoid overflow and underflow issues.
3. Your convergence criteria should be that the absolute change in EACH value of γ AND ϕ is less than ϵ (Use $\epsilon = 1e-3$).
4. You may want to check out the `psi` function in Matlab.