Homework 4
Graphical Models
Gibbs Sampling

CMU 10-715: Machine Learning (Fall 2015)
http://www.cs.cmu.edu/~bapoczos/Classes/ML10715_2015Fall/
OUT: Nov 9, 2015
DUE: Nov 23, 2015, 10:20 AM

Guidelines

- The homework is due at 10:20 am on Monday November 23, 2015. Each student will given two late days that can be spent on any homeworks, but at most one late day per homework. Once you have used up your late days for the term, late homework submissions will receive no credit.
- Submit both a paper copy and an electronic copy through through the submission website: https://autolab.cs.cmu.edu/courses/10715-f15. You can sign in using your Andrew credentials. You should make sure to edit your account information and choose a nickname/handle. This handle will be used to display your results for any competition style questions on the class leaderboard.
- Some questions will be *autograded*. Please make sure to carefully follow the submission instructions for these questions.
- We recommend that you typeset your solutions using software such as LaTeX.If you write, ensure your handwriting is clear and legible. The TAs will not invest undue effort to decrypt bad handwriting.
- Programming guidelines:
  - **Octave:** You must write submitted code in Octave. Octave is a free scientific programming language, with syntax similar to that of MATLAB. Installation instructions can be found on the Octave website. (You can develop your code in MATLAB if you prefer, but you *must* test it in Octave before submitting, or it may fail in the autograder.)
  - **Autograding:** This problem is autograded using the CMU Autolab system. The code which you write will be executed remotely against a suite of tests, and the results used to automatically assign you a grade. To make sure your code executes correctly on our servers, you should avoid using libraries which are not present in the *basic* Octave install.
  - **Submission Instructions:** For each programming question you will be given a function signature. You will be asked to write a single Octave function which satisfies the signature. In the code handout linked above, we have provided you with a single folder containing stubs for each of the functions you need to complete. *Do not modify the structure of this directory or rename these files.* Complete each of these functions, then compress this directory *as a tar file* and submit to Autolab online. You may submit code as many times as you like.
    When you download the files, you should confirm that the autograder is functioning correctly by compressing and submitting the directory of stubs provided. This should result in a grade of zero for all questions.
  - **SUBMISSION CHECKLIST**
    - Submission executes on our machines in less than 3 minutes.
    - Submission is smaller than 5000K.
    - Submission is a `.tar` file.
    - Submission returns matrices of the *exact* dimension specified.

1 LDA model and Gibbs Sampling [Eric; 65 pts]

In this model, you will infer topics from word documents using Gibbs Sampling and Latent Dirichlet Allocation (LDA). We will use the following terminology:

- T is the number of topics, n is the number of words, m is the number of documents, and $|V|$ is the total number of distinct words.
- w_i is the ith word, which comes from document d_i and is assigned topic z_i for $i \in [n]$.
- θ is a $m \times T$ matrix of parameters, for each document / topic combination. $\theta^{(d_i)}$ will denote the corresponding row for document d_i.
- ϕ is a $T \times |V|$ matrix of parameters, for each possible topic / word combination. $\phi^{(z_i)}$ will denote the corresponding row for topic z_i.

1.1 LDA

In LDA, the general setting is that we have a collection of documents that can be organized into various topics, and the distribution of the words depends on the topic of the document it comes from. First you will answer some straightforward questions about the model. Let T be the total number of topics.

1. (5pts) In this model we make several assumptions:
   - Given a topic z_i and matrix ϕ, let $\phi^{(z_i)}$ be the vector of probability masses of each word given a topic z_i. In other words, $w_i|z_i, \phi \sim \text{Multinomial}(\phi^{(z_i)})$.
   - The topic z_i assigned to each word w_i is dependent on the document it comes from. Let $\theta^{(d_i)}$ be the vector of probability masses of each topic given a document d_i. In other words, $z_i|\theta, d_i \sim \text{Multinomial}(\theta^{(d_i)})$.
   - Lastly, we assume a Dirichlet prior on the rows of θ and ϕ. In other words, $\theta^{(d_i)} \sim \text{Dirichlet}(\alpha)$ and $\phi^{(z_i)} \sim \text{Dirichlet}(\beta)$.

   Draw the corresponding graphical model for this problem.
2. (2pts) Rewrite the joint distribution of $p(\mathbf{w}, \mathbf{z}, \phi, \theta)$, using the conditional independence from the above generative process.

1.2 Gibbs Sampling

We would like to learn the posterior distribution of the latent variables given the observed variables. Specifically, we want to learn

$$p(\theta, \phi, \mathbf{z}|\mathbf{w}) = \frac{p(\theta, \phi, \mathbf{z}, \mathbf{w})}{p(\mathbf{w})}$$

However, this is in general intractable, which provides the motivation for using Gibbs sampling. For this problem, you will derive the Gibbs sampling rule for the latent topics z_i. For a review of the Gibbs sampling algorithm, see the 3rd set of recitation notes from 11/10.

1. (6pts) To run Gibbs sampling, we need an expression for $p(z_i|\mathbf{z_{-i}}, \mathbf{w})$. First, show that

$$p(z_i = j|\mathbf{z_{-i}}, \mathbf{w}) \propto \int p(w_i|\phi^{(z_i)}, z_i)p(\phi^{(z_i)}|\mathbf{z}_{-i}, \mathbf{w}_{-i})d\phi^{(z_i)} \int p(z_i|\theta^{(d_i)})p(\theta^{(d_i)}|\mathbf{z}_{-i})d\theta^{(d_i)}$$

2. (10pts) Let $\eta_{-i,j}^{(w_i)}$ be the number of word instances of w_i assigned to topic j excluding w_i, and let $\eta_{-i,j}$ be the total number of word instances assigned to topic j. Similarly, let $\eta_{-i,j}^{(d_i)}$ be the number of words in document d_i assigned to topic j excluding w_i, and let $\eta_{-i}^{(d_i)}$ be the number of words in document d_i excluding w_i. Compute each integral to show that

$$p(z_i|\mathbf{z}_{-i}, \mathbf{w}) \propto \frac{\eta_{-i,j}^{(w_i)} + \beta}{\eta_{-i,j} + |V|\beta} \cdot \frac{\eta_{-i,j}^{(d_i)} + \alpha}{\eta_{-i}^{(d_i)} + T\alpha}$$

   The following will be greatly helpful:

- The Dirichlet is a conjugate prior to the multinomial. In particular, if $p|\alpha \sim \text{Dirichlet}(\alpha)$ and $X|p \sim \text{Multinomial}(p)$, then $p|X \sim \text{Dirichlet}(\alpha + c)$ where c is a vector of counts of each observed category.
- The mean of a Dirichlet(α) is $\frac{\alpha}{\sum_k \alpha_k}$.

1.3 Implementation

We will use the following notation in the code handout:

- w, d, and z are $n \times 1$ vectors where the ith elements are w_i, d_i, z_i.
- word_topic_counts is a $|V| \times T$ matrix where the i, jth entry contains the number of times the ith vocabulary word has been assigned the jth topic.
- document_topic_counts is a $m \times T$ matrix where the i, jth entry contains the number of times a word in document i has been assigned the jth topic.
- word_topic_counts_i and document_topic_counts_i denote the same counts excluding w_i.

1. (8pts) Implement [word_topic_counts,document_topic_counts] = lda_counts (w,d,z,T), which computes the corresponding counts for all the words, documents, and topics as defined above.
2. (8pts) Implement [p] = lda_cond_T (w_i,d_i,word_topic_counts_i, document_topic_counts_i,T,alpha,beta), which computes the conditional distribution of $z_i|\mathbf{z}_{-i}, \mathbf{w}$ using w_i, d_i and the corresponding counts with w_i already removed.

   Hint: remember to normalize your calculated distribution, and ensure your returned vector is $T \times 1$.
3. (10pts) Implement [z] = lda_gibbs (w,d,z,T,alpha,beta,niters), which runs Gibbs sampling for niters iterations (one iteration is a single pass over the data), and returns the last sample z. Note that the function takes in initial values for z.

   Hint: recomputing the counts from scratch will be a costly operation. Since every Gibbs sampling update only changes one value of z_i at a time, you should simply maintain the correct counts after each change.

   Due to the random nature of Gibbs sampling, this function will not be autograded. It will still be graded, so include this in your autolab submission!
4. Run your Gibbs sampling routine on the provided dataset in reuters.mat, which consists of a collection of 100 Reuters articles. This might take a while, so test your function on a smaller subset before running it on the full dataset. Run your algorithm with $T = 20$ topics for 1000 iterations, with $\alpha = \beta = 1$. Do the following:

   - (6pts) For each topic, sort the words by frequency and print the print the top 5 words of each topic. Each $w_i \in$w is an index of the vocubulary into the vector vocab, so you can retrieve a vocabulary word w with vocab(w).
   - (4pts) Can you identify any of the topics? Identify 3 or more topics that you found to be most reasonably grouped, and include these in your writeup.

   Tip: Note that you can "restart" the Gibbs sampling procedure by simply providing the most recent topics z as the initial values, in case you want to run your 1000 iterations in smaller batches.
5. (6pts) Now that you have a sampling procedure for the $\mathbf{z}|\mathbf{w}$, how could you estimate this distribution? You do not need to write code for this problem, a brief explanation will suffice.

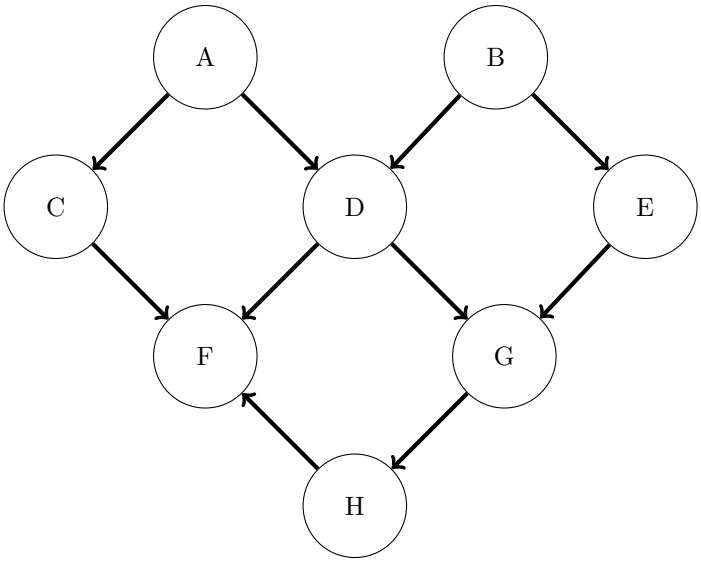

Figure 1: A graphical model.

2 Graphical Model and Inference [Fish; 35 pts]

1. (5pts) Are there any pairs of point that are independent? If you answer is yes, please list out all the pairs. Please provide reasons.

2. (5pts) Given D and A, are B and C independent? Please provide reasons.

3. (5pts) Given D and A, are B and F independent? Please provide reasons.

4. (5pts) Suppose H is your observation, and you want to infer the rest of the parameters using Gibbs sampling. What is the posterior probability that we want to get?

5. (5pts) Suppose we want to use Gibbs sampling to do the inference, write out the pseudo code for the algorithm. Your Input is $(a_0, b_0, c_0, d_0, e_0, f_0, g_0, h)$ and your output should be the estimated parameters $(\hat{A}, \hat{B}, \hat{C}, \hat{D}, \hat{E}, \hat{F}, \hat{G})$

6. (5pts) Using an undirected version of the model, write down the joint distribution over all of the variables. Use $\phi_{IJ}(I, J)$ to represent the joint distribution between node I and J.

7. (5pts) Are there any differences between the set of conditional independence encoded by the directed and undirected versions of this model? Please provide reasons.