Inference and Representation, Fall 2016

Problem Set 4: Gibbs sampling

Due: Monday, October 17, 2016 at 3pm (uploaded to Gradescope/NYU Classes.)

Your submission should include a PDF file called "solutions.pdf" with your written solutions, separate output files, and all of the code that you wrote.

Important: *See problem set policy on the course web site.*

---

1. **Conjugacy and Bayesian prediction** (generalization of Bernoulli example from Murphy 9.2.5.5):

   (a) Let $\theta \sim \mathrm{Dir}(\alpha)$. Consider discrete random variables $(X_1, X_2, \ldots, X_N)$, where $X_i \sim \mathrm{Cat}(\theta)$ for each i (thus the X_i are conditionally independent of one another given θ). Show that the posterior $\Pr(\theta \mid x_1, \ldots, x_N, \alpha)$ is given by $\mathrm{Dir}(\alpha')$, where

   $$\alpha'_k = \alpha_k + \sum_{i=1}^{N} 1[x_i = k].$$

   This property, that the posterior distribution $\Pr(\theta \mid \mathbf{x})$ is in the same family as the prior distribution $\Pr(\theta)$, is called *conjugacy*. The Dirichlet distribution (see Murphy Sec. 2.5.4) is the *conjugate prior* for the Categorical distribution. Every distribution in the exponential family has a conjugate prior. For example, the conjugate prior for the mean of a Gaussian distribution can be shown to be another Gaussian distribution.

   (b) Now consider a random variable $X_{\text{new}} \sim \mathrm{Cat}(\theta)$ that is assumed conditionally independent of $(X_1, X_2, \ldots, X_N)$ given θ. Compute:

   $$p(x_{\text{new}} \mid x_1, x_2, \ldots, x_N, \alpha)$$

   by integrating over θ.

   Hint: Your result should take the form of a ratio of gamma functions.

   This is called *Bayesian* prediction because we put a prior distribution over the parameters θ (in this case, a Dirichlet) and are thus able to take into consideration our initial uncertainty over (and prior knowledge of) the parameters together with the evidence we observed (samples $x_1, \ldots, x_N$) when giving our predictions for x_{new}.

2. Latent Dirichlet allocation (LDA) is a probabilistic model for discovering topics in sets of documents [1]. The generative model is as follows:

   - For each document, $m = 1, \ldots, M$
     (a) Draw topic probabilities $\theta_m \sim p(\theta|\alpha)$
     (b) For each of the N words:
        i. Draw a topic $z_{mn} \sim p(z|\theta_m)$
        ii. Draw a word $w_{mn} \sim p(w|z_{mn}, \beta)$,

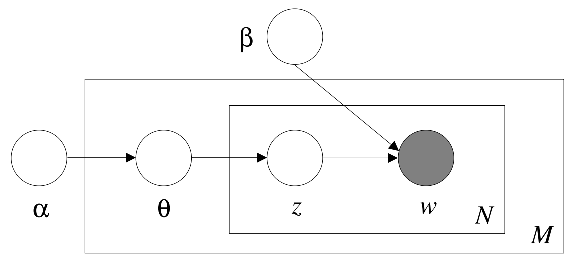

Figure 1: Graphical structure of the LDA model.

where $p(\theta|\alpha)$ is a Dirichlet distribution, and where $p(z|\theta_m)$ and $p(w|z_{mn}, \beta)$ are Multinomial distributions. Treat α and β as fixed hyperparameters. Note that β is a matrix, with one column per topic, and the Multinomial variable z_{mn} selects one of the columns of β to yield multinomial probabilities for w_{mn}.

(a) In this question you will use an off-the-shelf implementation of LDA to get practice with learning topic models on real-world data, and to analyze various trade-offs that can be made during learning.

   i. Prepare a corpus of documents from which you'll learn. You can find some already prepared text collections here:
   `https://archive.ics.uci.edu/ml/datasets/Bag+of+Words`
   However, we prefer that you be creative and construct your own!

   ii. Learn a latent Dirichlet allocation model on your corpus using default parameters. You can use any software package that you like. Two excellent options are:
   - Mallet (`http://mallet.cs.umass.edu/`)
   - Gensim (`http://radimrehurek.com/gensim/`)

   Qualitatively describe what topics are discovered.

   iii. Re-run learning using varying numbers of topics (e.g., 5, 20, 100). Describe qualitatively the differences that you observe as the number of topics increases.

(b) Derive a Gibbs sampler for the LDA model (i.e., write down the set of conditional probabilities for the sampler; see Sec. 24.2 of Murphy). To obtain full credit, you must hand in your full derivation, not just the final formulas.

   You may find it helpful to refer to your solutions from question 1.

(c) Derive a collapsed Gibbs sampler for the LDA model, where you consider the marginal distribution $\Pr(\mathbf{z}_m \mid \mathbf{w}_m; \alpha, \beta)$ (integrating out *just* the topic probabilities θ_m; here we assume that β is known) and are now only sampling $\mathbf{z}$. Again, you must hand in your full derivation.

(d) Implement both of the inference algorithms that you derived. You will then run your algorithms to find the posterior topic distribution θ for an input document.

   We have previously learned the parameters (i.e., α and β) of a 200-topic LDA model on a corpus containing thousands of abstracts of papers from the top machine learning conference, Neural Information Processing Systems (NIPS). Your task will be to infer the topic distribution for a new document.

   We have provided the following data files:

- alphas.txt, which has on each line for topic i: i, α_i, and a list of the most likely words for this topic,
- abstract_*.txt, with the words of document m (i.e., the abstract),
- abstract_*.txt.ready, with, in order,
  - the number of topics k,
  - α_i, for $i = 1, \ldots, k$,
  - for every word w_n, the word itself followed by $\beta_{w_n,i}$ for $i = 1, \ldots, k$.

Note that your code only needs to read in the abstract_*.txt.ready files – the alphas.txt and abstract_*.txt files are provided for your reference only.

It is common with MCMC methods to discard the first X samples to avoid using samples that are highly correlated with the arbitrary starting assignment (this is called "burning in"). Use $X = 50$ for your Gibbs sampling implementations.

For each of the abstracts,

i. Use your code to generate an accurate estimate of $E[\theta]$ using collapsed Gibbs sampling with a high number of iterations (e.g. 10^4). Use this as ground truth. The following formula can be used to obtain an estimate of θ from the collapsed Gibbs sampler (where T is the number of samples):

$$E[\theta_i] = \frac{T\alpha_i + \sum_{t=1}^{T} \sum_{n=1}^{N} 1[z_n^t = i]}{T(\sum_{\hat{i}=1}^{k} \alpha_{\hat{i}} + N)}$$

ii. Plot the ℓ_2 error on your estimate of $E[\theta]$ as a function of the number of iterations for each of the algorithms.

Only include in your solutions the plot for the data file NIPS2008_0517. The remaining files are provided for your own experimentation.

You may use the programming language of your choice. We recommend first checking that packages are available to (1) sample from a Dirichlet distribution, and (2) compute the Digamma function $\Psi(x)$, as these will simplify your coding. For example, see Python's numpy.random.mtrand.dirichlet and scipy.special.psi.

References

[1] David M. Blei, Andrew Ng, and Michael Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.