Latent Dirichlet Allocation

Subtitle

Cynthia Cook
cmc496@psu.edu

Advisor: David Hunter
dhunter@stat.psu.edu

Department of Statistics
Pennsylvania State University
December 2, 2015

Latent Dirichlet Allocation

Subtitle

Cynthia Cook

Abstract

Will include once I have a clear research plan...

1. Introduction

For the time being, I am studying the statistical properties of the Latent Dirichlet Allocation topic model.

2. Background on LDA

With an increase in the amount of digitized text, topic modeling has become an important area of research. One of the most popular methods for topic modeling is Latent Dirichlet Allocation (LDA) first described in [2]. LDA is a probabilistic hierarchical generative model, which uses the Bayesian framework to discover hidden structure, or *topics*, within a text Corpus. **For a more thorough background in topic modeling, refer to [4].**

2.1. Generative Process

Given a set of K topics $k = 1, ...K$ and a set of D documents $d = 1, ..., D$, also referred to as a corpus, the vocabulary is the set of unique words in the Corpus say $w_{1,1}, ..., w_{d,n_d}, ..., w_{D,n_D}$, where n_d is the number of uniques words in document d and $V = \sum_{d=1}^{D} n_d$. The goal of this algorithm as stated by Blei, "to find a probabilistic model of a corpus that not only assigns high probability to members of the corpus, but also assigns high probability to other 'similar' documents" ([2], pp.996). The generative process for a corpus is as follows:

- For each topic,
  1. Draw $\beta^k \sim Dir(\eta)$, a distribution for each topic over the vocabulary.
- For each document,
  2. Draw $\theta^d \sim Dir(\alpha)$, a distribution for each document over the topics.
  - For each word in document d,
    3. Draw $z_{d,n} \sim Mult(\theta^d)$, a topic assignment (i.e. $z_{d,n} \in \mathbb{Z} \in \{1, ..., K\}$).
    4. Draw $w_{d,n} \sim Mult(\beta^{z_{d,n}})$, the nth word in document d.

This process requires that we know the number of topics K a priori. Blei also assumed that the hyperparameters η and α are fixed and can be estimated through the given data. Assuming this model, we can define the joint probability distribution of a corpus as:

$$p(\beta, \theta, z_{d,n}, w_{d,n}|\alpha, \eta) = \prod_{k=1}^{K} p(\beta^k|\eta) \prod_{d=1}^{D} \left(p(\theta^d|\alpha) \prod_{n=1}^{n_d} p(z_{d,n}|\theta^d) p(w_{d,n}|\beta^1, ..., \beta^K, z_{d,n}) \right), \tag{1}$$

We will use the following dot notation:

$$c_{k,d,v} = \sum_{n=1}^{n_d} I(z_{d,n} = k \& w_{d,n} = v); \qquad \# \text{ of times word } v \text{ is assigned to topic } k \text{ in document } d$$

$$c_{k,.,v} = \sum_{d=1}^{D} c_{k,d,v}; \qquad \# \text{ of words in document } d \text{ assigned to topic } k$$

$$c_{k,d,.} = \sum_{v=1}^{V} c_{k,d,v}; \qquad \# \text{ of times word } v \text{ is assigned to topic } k \text{ in any document}$$

$$c_{k,.,.} = \sum_{d=1}^{D} \sum_{v=1}^{V} c_{k,d,v}; \qquad \# \text{ of words in corpus assigned to topic } k$$

Let $\Gamma(\cdot)$ be the Gamma function, then:

$$p(\beta^k|\eta) = \frac{\Gamma(\sum_{v=1}^{V} \eta_v)}{\prod_{v=1}^{V} \Gamma(\eta_v)} \prod_{v=1}^{V} (\beta_v^k)^{\eta_v - 1}$$

$$p(\theta^d|\alpha) = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} (\theta_k^d)^{\alpha_k - 1}$$

$$p(z_{d,n}|\theta) = \frac{1}{\prod_{k=1}^{K} c_{d,k,.}!} \prod_{k=1}^{K} (\theta_k^d)^{c_{d,k,.}}$$

$$p(w_{d,n}|\beta, z) = \frac{1}{\prod_{v=1}^{V} c_{k,.,v}!} \prod_{v=1}^{V} (\beta_v^{z_{d,n}})^{c_{k,.,v}}$$

2.2. Posterior Inferance

In practice, we do not know either β or θ. All we are given is a set of documents providing the $w_{d,n}$ parameters. Most often, we are interested in the latent variables β and θ. Rewriting 1

$$p(w, z, \theta, \beta|\alpha, \eta) = \left(\prod_{k=1}^{K} \frac{\Gamma(\sum \eta_v)}{\prod_{v=1}^{V} (\Gamma(\eta_v))} \prod_{v=1}^{V} (\beta_v^k)^{(\eta_v - 1 + c_{k,.,v})} \right) * \left(\prod_{d=1}^{D} \frac{\Gamma(\sum \alpha_k)}{\prod_{k=1}^{K} (\Gamma(\alpha_k))} \prod_{k=1}^{K} (\theta_k^d)^{(\alpha_k - 1 + c_{k,d,.})} \right) \tag{2}$$

Integrating out the latent variables, we obtain the following posterior distribution of a corpus:

$$p(w|\theta, \beta, z) = \int_{\theta} \int_{\beta} \sum_{z} \left(\prod_{k=1}^{K} \frac{\Gamma(\sum \eta_v)}{\prod_{v=1}^{V} (\Gamma(\eta_v))} \prod_{v=1}^{V} (\beta_v^k)^{\eta_v - 1 + c_{k,.,v}} \right) * \left(\prod_{d=1}^{D} \frac{\Gamma(\sum \alpha_k)}{\prod_{k=1}^{K} (\Gamma(\alpha_k))} \prod_{k=1}^{K} (\theta_k^d)^{\alpha_k - 1 + c_{k,d,.}} \right) d\beta d\theta \tag{3}$$

This distribution can be used for maximum likelihood estimation and to find the distribution of the latent variables. However, it has been proven to be intractable [2]. The development of several fast and accurate approximation methods applied to LDA has provided a user with several options including collapsed Gibbs sampling [7], collapsed variational Bayesian inference [11], maximum likelihood estimation [9], and maximum a posteriori estimation [6]. We will explore both Gibbs sampling and variational methods in the following two subsections.

2.2.1. Gibbs Sampling

Gibbs sampling is a special form of the Metropolis Hasting MCMC algorithm. The idea, as applied to lda, is to approximate 3 by repeatedly sampling from the conditional distributions of the latent variables given all other known and estimated variables at each step. It is known that the Gibbs sampler constitutes a

Markov Chain whose stationary distribution is the function in question. The Markov Chain described below asymptotically converges to 3. Pseudo Code:

1. Initialize $\beta^{(0)}, \theta^{(0)}$, and all of the topic assignments $z^{(0)}$ for each word in the vocabulary.
2. For t in $1, 2, ...,$ sample:
   (a) $z_i^{(t)}$ from $p(z_i^{(t)}|z_{-i}^{(t-1)}\beta^{(t)}, \theta^{(t-1)}, w, \eta, \alpha)$
   (b) $\beta^{(t)}$ from $p(\beta^{(t)}|\theta^{(t-1)}, z^{(t-1)}, w, \eta, \alpha) = Dir(\eta + c_{k,.,v})$
   (c) $\theta^{(t)}$ from $p(\theta^{(t)}|\beta^{(t)}, z^{(t)}, w, \eta, \alpha) = Dir(\alpha + c_{k,d,.})$
3. Continue to update and sample each of the latent variables until convergence.

Although this algorithm will eventually converge to the distributions of the latent variables, it is slow. After noticing that the distribution of topic probabilities for each word is independent of both θ and β, a collapsed gibbs sampler can be implemented. Here, the algorithm integrates out both θ and β from the Markov Chain's state space. Then it uses a Gibbs sampler to iteratively sample the topic assignments. After solving for converged estimates, both θ and β can be easily estimated. For more information on the collapsed gibbs sampler, refer to [10]. After integrating out these variables, [12] showed that for word n in document d, the probability of word $w_{d,n}$ being assigned topic $z_{d,n} = k$ is

$$p(z_{d,n}^t = k|z_{-(d,n)}^{(t-1)}, \theta, \beta, w) = c^{-1} * \frac{(c_{z_{d,n},d,.}^{-(d,n)} + \alpha_{z_{d,n}}) * (c_{z_{d,n},.,w_{d,n}}^{-(d,n)} + \beta_{w_{d,n}})}{c_{z_{d,n},.,.}^{-(d,n)} + \sum_{v=1}^{V} \beta_v} \tag{4}$$

where c is the normalizing constant equal to

$$\sum_{k=1}^{K} \frac{(c_{k,d,.}^{-(d,n)} + \alpha_k) * (c_{k,.,w_{d,n}}^{-(d,n)} + \beta_{w_{d,n}})}{c_{k,.,.}^{-(d,n)} + \sum_{v=1}^{V} \beta_v}$$

The update equations for θ and β are as follows [12]:

$$\hat{\theta}_{d,k} = \frac{\alpha_k + n_{d,k,.}}{D\alpha + n_{.,d,.}} \qquad \hat{\beta}_{v,k} = \frac{\beta_{k,v} + n_{k,.,v}}{J\beta + n_{k,.,.}} \tag{5}$$

Psuedo Code for the collapsed Gibbs sampler is as follows:

1. Initialize the topic assignments for each word in the vocabulary $z^{(0)}$.
2. For each w_i and $t = 1, 2, 3, ...$:
   (a) For each topic $k = 1, 2, 3, ...$ compute 4.
   (b) Draw the new topic assignment from the computed discrete distribution $z_i^{(t)}$.
3. Continue until convergence.
4. Calculate $\hat{\theta}$ and $\hat{\beta}$ from 5.

2.2.2. Variational Bayes

In order to infer the parameters, a second approach using variational methods was first described in [2]. This approach uses Jensen's inequality to obtain a lower bound on the log likelihood of the data given α, η: $l(\alpha, \eta) = \sum_{d=1}^{D} (\log (w_d|\alpha, \eta))$. In general, a family of tractable surrogate distributions defined by their unique parameters is found. These parameters are referred to as variational parameters. The lower bound is found by minimizing the Kulllback-Leibler (KL) divergence between the variational distribution and the true posterior. A simple variational distribution to consider, the one that Blei considers, just removes the dependence between θ and β, which causes the intractability of the true posterior. Here we will use the following surrogate function:

$$\tilde{q}(z, \beta, \theta) = \prod_k \tilde{q}(\beta_k|\tilde{\eta}_k) \prod_d \tilde{q}(\theta_d|\tilde{\alpha}_d) \prod_{d,n} \tilde{q}(z_{d,n}|\tilde{\gamma}_{d,n}).$$

Minimizing the KL distance leads to the following update equations:

$$\tilde{\gamma}_{n,d,k}^{(t+1)} = \exp\left(\Psi(\tilde{\alpha}_{d,k}^{t}) + \Psi(\tilde{\eta}_{k,x_{d,n}}^{t}) - \Psi\left(\sum_{n}(\tilde{\eta}_{k,n}^{t})\right)\right) \qquad \tilde{\alpha}_{d,k}^{t} = \alpha + \sum_{n}(\tilde{\gamma}_{d,k,n}^{(t+1)}) \qquad \tilde{\eta}_{k,n}^{(t+1)} = \eta + \sum_{d,n}(I(x_{d,n} = w)\tilde{\gamma}_{d,k,n}^{t}) \tag{6}$$

where $\Psi(x) = \frac{d\log(\Gamma x)}{dx}$ and I is the indicator function. Blei goes a step further and proposes the following EM algorithm to estimate the β and α parameters:

1. (E-step) Find the variational parameters by using a fixed point iterative algorithm to minimize the KL distance.
2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to β and α.

For the M-step, β can be found analytically using

$$\beta^{(t+1)} \propto \sum_{d=1}^{D}\sum_{n=1}^{N_d}(\tilde{\eta}_{d,k,n}w_{d,k,n}), \tag{7}$$

but $\alpha^{(t+1)}$ must be found using a Newton Raphson algorithm. Here, the estimate for β is proportional because we need to normalize so that the β's sum to one.

Below outlines Pseudo Code for the variational Bayes approximation algorithm:

1. Initialize:

$$\begin{aligned} \text{For each topic } k, &\qquad \tilde{\eta}_n^0 = \eta_n + fracNk \\ \text{For each document } d, &\qquad \tilde{\alpha}_k^0 = \alpha_k + fracNk \\ \text{For each word in document } d, &\qquad \tilde{\gamma}_k^0 = frac1k \end{aligned}$$

2. (E-step) For each document d in $1, ..., D$,
   Repeat for $t = 0, 1, 2, ...$:
   (a) For n in $1, ..., N_d$
   (b) For k in $1, ..., K$
   (c) Update $\tilde{\gamma}_{n,d,k}^{(t+1)}$ using 6.
   (d) Normalize $\tilde{\gamma}_{n,d}^{(t+1)}$ to sum to one.
   (e) Update $\tilde{\alpha}^{(t+1)}$ and $\tilde{\eta}^{(t+1)}$ using 6
   (f) Continue until convergence.
3. Solve the log likelihood given the data and variational estimates $\tilde{\alpha}$ and $\tilde{\eta}$.
4. (M-step) For each document d in $1, ..., D$
5. For each k in $1, ..K$
6. For each n in $1, ..., N_d$
7. Solve for $\beta_{n,k}$ using 7
8. Normalize the β so they sum to one.
9. Solve for α.
10. If likelihood converged using new parameters then end.
11. Else go back to the E-step.

1.

3. Implementation/Data Example

In [2], variational methods are applied to solve the problem. These methods were coded in C by Blei (refer to [3]) and have since been utilized in the *topicmodels* package in R [8]. Another package, *lda*, in R has been developed by Chang, which uses his own C code with collapsed Gibbs sampling for posterior approximation [5]. On the website Cross Validated, there is a discussion of which package is better for users, with no definitive answer [1]. Refer to my 540 project and what I find there...

3.1. Data

What and how I got it...

3.2. lda results

4. Conclusions

[1] Anonymous. (2012), "Two R Packages for Topic Modeling, lda and topicmodels?" *Cross Validated*: http://stats.stackexchange.com/questions/24441/two-r-packages-for-topic-modeling-lda-and-topicmodels.

[2] Blei, D., Ng, A., and Jordan, M. (2003), "Latent Dirichlet Allocation." *Journal of Machine Learning Research*: 3 993-1022.

[3] Blei, D. (2004),"LDA-C."

[4] Blei, D. (2012), "Survaying a Suit of Algorithms that offer a solution to managing large document archives: Probalistic Topic Models." *Communications fo the ACM*: vol.55, no.4, 77-84.

[5] Chang, J. (2015), "Package 'lda'." *CRAN*.

[6] Chien, J., and Wu, M. (2008), "Adaptive Bayesian Semantic Analysis." *Audio, Speech, and Language Processing*, IEEE Transactions on: 16(1), 198-207.

[7] Griffiths, L. and Steyvers, M. (2004), "Finding Scientific Topics." *PNAS*: 1(Suppl 1), 5228-5235.

[8] Grun, B. and Hornik, K. (2015), "Package 'topicmodels'." *CRAN*.

[9] Hofmann, T. (2001), "Unsupervised Learning by probabilistic Latent Semantic Analysis." *Machine Learning*: 42(1), 177-196.

[10] Liu, Jun. (1994), "The Collapsed Gibbs Sampler in Bayesian Computations with Applications to a Gene Regulation Problem." *ASA Jorunal*: Vol.89, No.427 Theory and Methods, 958-966.

[11] Teh, Y. W., Newman, D., and Welling, M. (2007), "A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation."*NIPS*: 3, 1353-1360.

[12] Carpenter, Bob. (2010), "Integrating out Multinomial Parameters in Latent Dirichlet Allocation and Naive Bayes for Collapsed Gibbs Sampling." *LingPipe*: carp@lingpipe.com.