Latent Dirichlet Allocation: A Student Companion

Colorado Reed

January 17, 2012

Abstract

The aim of this tutorial is to introduce the reader to Latent Dirichlet Allocation (LDA) for topic modeling. This tutorial is not all-inclusive and should be accompanied/cross-referenced with Blei et al. (2003). The unique aspect of this tutorial is that I provide a full pseudo-code implementation of variational expectation-maximization LDA and an R code implementation at `http://highenergy.physics.uiowa.edu/~creed/#code`. The R code is arguably the simplest variational expectation-maximization LDA implementation I've come across. Unfortunately, the simple implementation makes it very slow and unrealistic for actual application, but it's designed to serve as an educational tool.

Contents

1 Prerequisites

This tutorial is most useful if you have the following background:

- basic background in probability, statistics, and inference, i.e. understand Bayes' rule and the concept of statistical inference
- understand the Dirichlet distribution
- understand the relationship between graphical models and statistical inference
- understand the expectation-maximization (EM) algorithm
- familiarity with the Kullback-Leibler (KL) divergence will be moderately helpful

If you do not have some or all of the above background, this tutorial can still be helpful. In the text I mention specific resources the interested reader can use to acquire this background.

2 Introduction

In many different fields we are faced with a ton of information: think Wikipedia articles, blogs, Flickr images, astronomical survey data, <insert some problem from your area of research here>, and we need algorithmic tools to organize, search, and understand this information. Topic modeling is a method for analyzing large quantities of unlabeled data. For our purposes, a **topic** is a probability distribution over a collection of words and a **topic model** is a formal statistical relationship between a group of observed and latent (unknown) random variables that specifies a probabilistic procedure to generate the topics—a *generative model*. The central goal of a topic is to provide a "thematic summary" of a collection of documents. In other words, it answers the question: what themes are these documents discussing? A collection of news articles could discuss e.g. political, sports, and business related themes.

3 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is arguable the most popular topic model in application. Let's examine the generative model for LDA, then I'll discuss inference techniques and provide some [pseudo]code and simple examples that you can try in the comfort of your home.

3.1 Higher-level Details

First and foremost, LDA provides a generative model that describes how the documents in our dataset were created.[1] Our dataset is a collection of D documents. But what is a document? It's a collection of words. So our generative model describes how each document obtains its words. Initially, let's assume we know K topic distributions for our dataset, meaning K multinomials containing V elements each, where V is the number

[1] Not literally, of course, this is a simplification of how the documents were actually created.

of terms in our corpus: $\beta_{1:K}$; $|\beta_i| = V$. Given these distributions, the LDA generative process is as follows:

1. For each document:
   (a) randomly choose a distribution over topics (a multinomial of length K)
   (b) for each word in the document:
      (i) Probabilistically draw one of the K topics from the distribution over topics obtained in (a), say topic β_j
      (ii) Probabilistically draw one of the V words from β_j

This generative model emphasizes that documents contain multiple topics. For instance, a health article might have words drawn from the topic related to seasons such as *winter* and words drawn from the topic related to illnesses, such as *flu*. Step (a) reflects that each document contains topics in different proportion, e.g. one document may contain a lot of words drawn from the topic on seasons and no words drawn from the topic about illnesses, while a different document may have an equal number of words drawn from both topics. Step (ii) reflects that each individual word in the document is drawn from one of the K topics in proportion to the document's distribution over topics as determined in Step (i). The actual word selection depends on the the distribution over the V words in our vocabulary as determined by the selected topic, β_j. Note that the generative model does not make any assumptions about the order of the words in the documents, this is known as the *bag-of-words assumption*.

The central goal of topic modeling is to automatically discover the topics from a collection of documents. Therefore our assumption that we know the K topic distributions is not very helpful; we must learn these topic distributions. This is accomplished through statistical inference, and I will discuss some of these techniques in the next section. Figure 1 visually displays the difference between a generative model (what I just described) and statistical inference (the process of learning the topic distributions).

3.2 Formal details and LDA inference

To formalize LDA, let's first restate the generative process in more detail (compare with the previous description):

1. For each document:
   (a) draw a topic distribution, $\theta_d \sim \text{Dir}(\alpha)$, where $\text{Dir}(\cdot)$ is a draw from a uniform Dirichlet distribution with scaling parameter α
   (b) for each word in the document:
      (i) Draw a specific topic $z_{d,n} \sim \text{multi}(\theta_d)$ where $\text{multi}(\cdot)$ is a multinomial
      (ii) Draw a word $w_{d,n} \sim \beta_{z_{d,n}}$

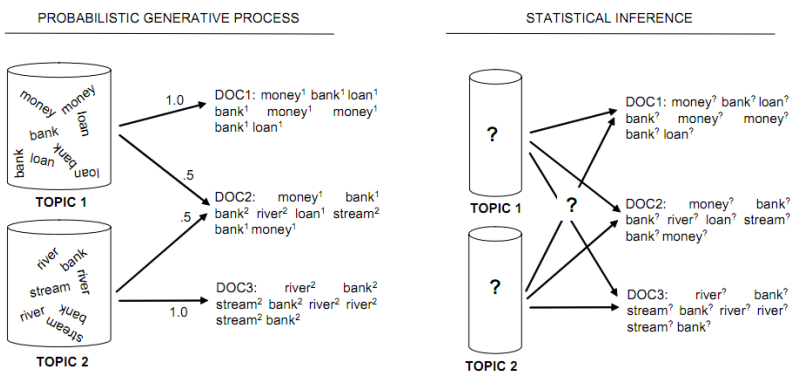

Figure 1: **Left**: a visualization of the probabilistic generative process for three documents, i.e. DOC1 draws from Topic 1 with probability 1, DOC2 draws from Topic 1 with probability 0.5 and from Topic 2 with probability 0.5, and DOC3 draws from Topic 2 with probability 1. The topics are represented by $\beta_{1:K}$ (where $K = 2$ in this case) in Figure 2 and the topic distributions for the two topics ($\{1, 0\}, \{0.5, 0.5\}, \{0, 1\}$) are represented by θ_d. **Right**: In the inferential problem we are interested in *learning* the topics and topic distributions. Image taken from Steyvers and Griffiths (2007).

Figure 2 displays the graphical model describing this generative process. A draw from a k dimensional Dirichlet distribution returns a k dimensional multinomial, θ in this case, where the k values must sum to one. The normalization requirement for θ ensures that θ lies on a $(k-1)$ dimensional simplex and has the probability density:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1}. \tag{1}$$

If you are not familiar with the Dirichlet distribution and Dirichlet sampling techniques, I encourage you to read Frigyik et al. (2010).

For reference purposes, let's formalize some notation before moving on:

- w represents a word and w^v represents the vth word in the vocabulary where $w^v = 1$ and $w^u = 0$ if the $v \neq u$—this superscript notation will be used with other variables as well.
- $\mathbf{w}$ represents a document (a vector of words) where $\mathbf{w} = (w_1, w_2, \ldots, w_N)$
- α is the parameter of the Dirichlet distribution, technically $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k)$, but unless otherwise noted, all elements of α will be the same, and so in the included R code α is simply a number.
- $\mathbf{z}$ represents a vector of topics, where if the ith element of $\mathbf{z}$ is 1 then w draws from the ith topic

- β is a $k \times V$ word-probability matrix for each topic (row) and each term (column), where $\beta_{ij} = p(w^j = 1|z^i = 1)$

The central inferential problem for LDA is determining the posterior distribution of the latent variables given the document:

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{p(\mathbf{w}|\alpha, \beta)}.$$

This distribution is the crux of LDA, so let's break this down step by step for each individual document (the probability of the entire corpus is then acquired by multiplying the individual document probabilities—this assumes independence in the document probabilities). First, we can decompose the numerator into a hierarchy by examining the graphical model:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\mathbf{w}|\mathbf{z}, \beta)p(\mathbf{z}|\theta)p(\theta|\alpha)$$

It should be clear that $p(\mathbf{w}|\mathbf{z}, \beta)$ represents the probability of observing a document with N words given the a topic vector of length N that assigns a topic to each word from the $k \times V$ probability matrix β. So we can decompose this probability into each individual word probability and multiply them together yielding:

$$p(\mathbf{w}|\mathbf{z}, \beta) = \prod_{n=1}^{N} \beta_{z_n, w_n}$$

Next, $p(\mathbf{z}|\theta)$ is trivial once we note that $p(z_n|\theta) = \theta_i$ such that $z_n^i = 1$, since after all, θ is just a multinomial. Finally, $p(\theta|\alpha)$ is given by Eq. (1). Bringing this all together we have:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = \left(\frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1} \right) \prod_{n=1}^{N} \beta_{z_n, w_n} \theta_{z_n}$$

where I am using θ_{z_n} to represent the component of θ chosen for z_n. Let's rephrase this probability using the superscript notation mentioned above, and for convenience we'll use the entire vocabulary of size V when calculating the probability and rely on an exponent to ween out the words that are used for each document:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = \left(\frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1} \right) \prod_{n=1}^{N} \prod_{i=1}^{k} \prod_{j=1}^{V} (\theta_i \beta_{i,j})^{w_n^j z_n^i} \tag{2}$$

We marginalize over θ and $\mathbf{z}$ in order to obtain the denominator (often referred to as the *evidence*) of Eq. (3.2):

$$p(\mathbf{w}|\alpha, \beta) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \int \left(\prod_{i=1}^{k} \theta_i^{\alpha_i - 1} \right) \left(\prod_{n=1}^{N} \sum_{i=1}^{k} \prod_{j=1}^{V} (\theta_i \beta_{ij})^{w_n^j} \right) d\theta$$

Notice this expression is the same as Eq. (2) except we integrate over θ and sum over $\mathbf{z}$. Unfortunately, computing this distribution is intractable as the coupling between θ and β makes it so that when we compute the log of this function we are unable to separate the θ and β. So while exact inference is not tractable, various approximate inference techniques can be used. Here I examine variational inference in some detail.

3.2.1 Variational Inference for LDA

The essential idea of variational inference is to use a simpler, convex distribution that obtains an adjustable lower bound on the log likelihood of the actual distribution. *Variational parameters* describe the family of simpler distributions used to determine a lower bound on the log likelihood and are optimized to create the tightest possible lower bound.

As discussed in Blei et al. (2003), an easy way to obtain a tractable family of lower bounds is to modify the original graphical model by removing the troublesome edges and nodes. In the LDA model in Figure 2 the coupling between θ and β makes the inference intractable. By dropping the problematic edges and nodes we obtain the simplified graphical model in Figure 3. This variational distribution has a posterior for each document in the form:

$$q(\theta, \mathbf{z}|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n). \tag{3}$$

The next step is to formally specify an optimization problem to determine the values of γ and ϕ. For the sake of brevity I must defer the reader to A.3 of Blei et al. (2003) for derivation of the following results. In particular, finding an optimal lower bound on the log likelihood results in the following optimization problem:

$$(\gamma^*, \phi^*) = \arg\min_{(\gamma,\phi)} \mathrm{D}\left(q(\theta, \mathbf{z}|\gamma, \phi)||p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)\right) \tag{4}$$

which is a minimization of the Kullback-Leibler (KL) divergence between the variational distribution and the actual posterior distribution. The KL divergence between the variational and actual distribution is the name of the game for variational Bayesian inference, and if you are not already familiar with KL divergence or the basics of variational inference, I encourage you to read sections 1.6 and 10.1 of Bishop (2006), respectively.

One method to minimize this function is to use an iterative fixed-point method, yielding update equations of:

$$\phi_{ni} \propto \beta_{iw_n} \exp\left\{\mathbb{E}_q[\log(\theta_i)|\gamma]\right\} \tag{5}$$

$$\gamma_i = \alpha_i + \sum_{n=1}^{N} \phi_{ni}. \tag{6}$$

where as shown in Blei et al. (2003) the expectation in the ϕ update is computed as

$$\mathbb{E}_q[\log(\theta_i)|\gamma] = \Psi(\gamma_i) - \Psi(\sum_{j=1}^{k} \gamma_j)$$

where Ψ is the first derivative of the logΓ function, which is "simply" this crazy looking

numerical approximation (computed via a Taylor approximation):

$$\Psi(x) \approx \left(\left((0.00418\bar{3}\frac{1}{(x+6)^2} - 0.003968)\frac{1}{(x+6)^2} + 0.008\bar{3}\right)\frac{1}{(x+6)^2} - 0.08\bar{3}\right)\frac{1}{(x+6)^2} + \log(x) - \frac{1}{2x} - \sum_{i=1}^{6}\frac{1}{x-i}$$

I've just bustled through quite a few equations, so let's pause for moment and gain some intuition on these results. The Dirichlet update in Eq. (6) is a posterior Dirichlet given the expected observations taken under the variational distribution, $\mathbb{E}[z_n|\phi_n]$. In other words, we're iteratively updating the variational Dirichlet topic parameter, γ, using the multinomial that best describes the observed words. The multinomial update essentially uses Bayes' theorem, $p(z_n|w_n) \propto p(w_n|z_n)p(z_n)$, where $p(w_n|z_n) = \beta_{iw_n}$ and $p(z_n)$ is approximated by the exponential of the expectation of its logarithm under the variational distribution. Finally, be sure to note the the variational parameters are actually a function of $\mathbf{w}$ although this was not explicitly expressed in the above equations. The optimization of Eq. (4) relies on a specific $\mathbf{w}$, see A.3 of Bishop (2006).

The fastidious reader may be wondering: "but how in the world do we find β or α? The previous optimization assumed we knew these parameters..." The answer is that we estimate β and α assuming we know ϕ and γ. Hopefully this approach sounds familiar to you: it's the Expectation-Maximization (EM) algorithm on the variational distribution! If you need a brief refresher on the EM algorithm please consult the appendix. If you need to learn the EM algorithm, please see sections 9.2-9.4 of Bishop (2006).

In the E-step of the EM algorithm we determine the log likelihood of the complete data—in other words, assuming we know the hidden parameters, β and α in this case. In the M-step we maximize the lower bound on the log likelihood with respect to α and β. More formally:

- **E-Step**: Find the optimal values of the variational parameters γ_d^* and ϕ_d^* for every document in the corpus. Knowing these parameters allows us to compute the expectation of the log likelihood of the complete data
- **M-Step**: Maximize the lower bound on the log likelihood of

  $$\ell(\alpha, \beta) = \sum_{d=1}^{M} \log p(\mathbf{w}_d|\alpha, \beta)$$

  with respect to α and β. This corresponds to finding maximum likelihood estimates with expected sufficient statistics for each document.

Here I quote the results for the M-step update but encourage the curious reader to peruse A.3 and A.4 of Blei et al. (2003) for derivations—it involves some neat tricks to make the inference scalable.

$$\beta_{ij} \propto \sum_{d=1}^{M}\sum_{n=1}^{N_d} \phi_{dni}^* w_{dn}^j \tag{7}$$

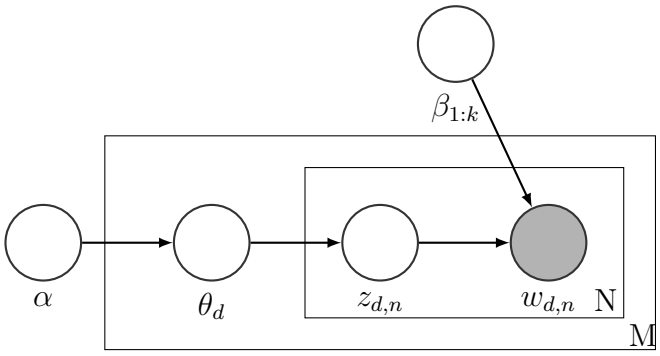

Figure 2: LDA graphical model

Note that the "proportional to" symbol ($\propto$) in the description of β_{ij} simply means that we normalize all β_i to sum to one. The α update is a bit trickier and uses a linear-scaling Newton-Rhapson algorithm to determine the optimal alpha, with updates carried out in log-space (assuming a uniform α):

$$\log(\alpha^{t+1}) = \log(\alpha^t) - \frac{\frac{dL}{d\alpha}}{\frac{d^2L}{d\alpha^2}\alpha + \frac{dL}{d\alpha}} \tag{8}$$

$$\frac{dL}{d\alpha} = M(k\Psi^{'}(k\alpha) - k\Psi^{'}(\alpha)) + \sum_{d=1}^{M}\left(\Psi(\gamma_{di}) - \Psi\left(\sum_{j=1}^{k}\gamma_{dj}\right)\right) \tag{9}$$

$$\frac{d^2L}{d\alpha^2} = M(k^2\Psi^{''}(k\alpha) - k\Psi^{''}(\alpha)) \tag{10}$$

This completes the nitty gritty of variational inference with LDA. Algorithm 1 provides the psuedocode for this variational inference and accompanying R code can be found at `http://highenergy.physics.uiowa.edu/~creed/#code`.

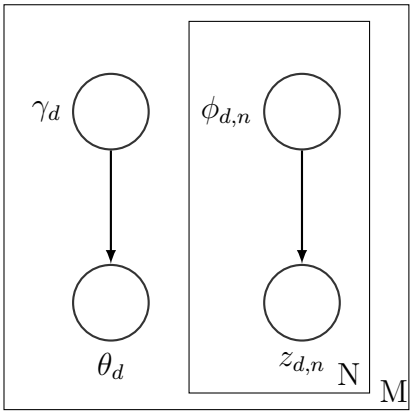

Figure 3: Variational distribution used to approximate the posterior distribution in LDA, γ and ϕ are the variational parameters.

Algorithm 1: Variational Expectation-Maximization LDA

Input: Number of topics K
Corpus with M documents and N_d words in document d
Output: Model parameters: β, θ, z

initialize $\phi^0_{ni} := 1/k$ for all i in k and n in N_d
initialize $\gamma_i := \alpha_i + N/k$ for all i in k
initialize $\alpha := 50/k$
initialize $\beta_{ij} := 0$ for all i in k and j in V

//E-Step (determine ϕ and γ and compute expected likelihood)
loglikelihood := 0
for $d = 1$ **to** M
  repeat
    for $n = 1$ **to** N_d
      for $i = 1$ **to** K
        $\phi^{t+1}_{dni} := \beta_{iw_n} \exp\left(\Psi(\gamma^t_{di})\right)$
      endfor
      normalize ϕ^{t+1}_{dni} to sum to 1
    endfor
    $\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \phi^{t+1}_{dn}$
  until convergence of ϕ_d and γ_d
  loglikelihood := loglikelihood + $L(\gamma, \phi; \alpha, \beta)$ // See equation BLAH
endfor

//M-Step (maximize the log likelihood of the variational distribution)
for $d = 1$ **to M**
  for $i = 1$ **to K**
    for $j = 1$ **to V**
      $\beta_{ij} := \phi_{dni} w_{dnj}$
    endfor
    normalize β_i to sum to 1
  endfor
endfor
estimate α via Eq. (8)

if loglikelihood converged **then**
  return parameters
else
  go back to E-step
endif

4 But how does LDA work?

Figure 4 shows example topics obtained using LDA on the Touchstone Applied Science Associates (TASA) corpus (a collection of nearly 40,000 short documents from educational writings). Each topic includes the 16 words with the highest probability for the topic: the largest β value for a given topic. A natural question to ask is, "how does LDA work?" or "why does LDA produce groups of words with similar themes?" For instance, with the four provided topics we note that the left-most topic is related to drugs and medicine, the next topic is related to color, the third topic is related to memory, and the fourth topic is related to medical care. How did LDA extract these topics from a collection of texts? The simple and straightforward answer is *co-occurance*. LDA extracts clusters of co-occurring words to form topics.

Topic 247

word	prob.
DRUGS	.069
DRUG	.060
MEDICINE	.027
EFFECTS	.026
BODY	.023
MEDICINES	.019
PAIN	.016
PERSON	.016
MARIJUANA	.014
LABEL	.012
ALCOHOL	.012
DANGEROUS	.011
ABUSE	.009
EFFECT	.009
KNOWN	.008
PILLS	.008

Topic 5

word	prob.
RED	.202
BLUE	.099
GREEN	.096
YELLOW	.073
WHITE	.048
COLOR	.048
BRIGHT	.030
COLORS	.029
ORANGE	.027
BROWN	.027
PINK	.017
LOOK	.017
BLACK	.016
PURPLE	.015
CROSS	.011
COLORED	.009

Topic 43

word	prob.
MIND	.081
THOUGHT	.066
REMEMBER	.064
MEMORY	.037
THINKING	.030
PROFESSOR	.028
FELT	.025
REMEMBERED	.022
THOUGHTS	.020
FORGOTTEN	.020
MOMENT	.020
THINK	.019
THING	.016
WONDER	.014
FORGET	.012
RECALL	.012

Topic 56

word	prob.
DOCTOR	.074
DR.	.063
PATIENT	.061
HOSPITAL	.049
CARE	.046
MEDICAL	.042
NURSE	.031
PATIENTS	.029
DOCTORS	.028
HEALTH	.025
MEDICINE	.017
NURSING	.017
DENTAL	.015
NURSES	.013
PHYSICIAN	.012
HOSPITALS	.011

Figure 4: Taken from Steyvers and Griffiths (2007).

Let's probe a bit deeper than simply saying LDA works because of co-occurance. Let's figure out why. To do this, we must reexamine the posterior distribution and break down what each member of the hierarchy implies:

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) \propto \underbrace{p(\mathbf{w}|\mathbf{z}, \beta)}_{1} \underbrace{p(\mathbf{z}|\theta)}_{2} \underbrace{p(\theta|\alpha)}_{3} \tag{11}$$

1. implies that making β a sparse matrix will increase the probability of certain words– remember that the β values for a given topic must sum to one so the more terms we assign a non-zero β value the thinner we have to spread our probability for the topic.

2. implies that making θ have concentrated components will increase the probability

3. implies that using a small α will increase the probability

The three factors form an interesting dynamic: (1) implies that having sparsely distributed topics can result in a high probability for a document, where the ideal way to form the sparse components is to make them non-overlapping clusters of *co-occurring* words in

different documents (why is this?); (2) encourages a sparse θ matrix so that the probability of choosing a given z value will be large, e.g. $\theta = (0.25, 0.25, 0.25, 0.25)$ would yield smaller probabilities than $\theta = (0.5, 0.5, 0, 0)$. In other words, (2) penalizes documents for having too many possible topics. (3) again penalizes using a large number of possible topics for a given document—small α values yield sparse θs. In summary: (1) wants to form sparse, segregated word clusters, (2) and (3) want to give a small number of possible topics for each document. But if we only have a few topics to choose from and each topic has a small number of non-zero word probabilities, then we surely better form meaningful clusters that could represent a diverse number of documents. How should we do this you ask? Form clusters of co-occurring terms, which is essentially what LDA accomplishes.

5 Further Study

- Blei et al. (2003): The original LDA paper; much of this tutorial was extracted from this paper. This paper also includes several excellent examples of LDA in application.
- Blei (2011): A recent overview and review of topic models, also includes directions for future research
- Steyvers and Griffiths (2007) provides a well-constructed introduction to probabilistic topic models and details on using topic models to compute similarity between documents and similarity between words.
- Dave Blei's video lecture on topic models: `http://videolectures.net/mlss09uk_blei_tm`

References

C.M. Bishop. *Pattern recognition and machine learning*, volume 4. 2006.

D. Blei. Introduction to probabilistic topic models. *Communications of the ACM*, 2011.

D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

BA Frigyik, A. Kapila, and M.R. Gupta. Introduction to the dirichlet distribution and related processes. *Department of Electrical Engineering, University of Washignton, UWEETR-2010-0006*, 2010.

M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440, 2007.

6 Appendix: EM Algorithm Refresher

The General EM Algorithm Summary:
Given a joint distribution $p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ over observed variable $\mathbf{X}$ and latent variables $\mathbf{Z}$, with parameters $\boldsymbol{\theta}$, the goal is to maximize the likelihood function $p(\mathbf{X}|\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$.

1. Initialize the parameters $\boldsymbol{\theta}^{old}$
2. **E Step** Construct $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{old}) \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

   which is the conditional *expectation* of the complete-data log-likelihood.

3. **M Step** Evaluate $\boldsymbol{\theta}^{new}$ via

$$\boldsymbol{\theta}^{new} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) \tag{12}$$

4. Check log likelihood and parameter values for convergence, if not converged let $\boldsymbol{\theta}^{old} \leftarrow \boldsymbol{\theta}^{new}$ and return to step 2.