Journal of Machine Learning Research 3 (2003) 993-1022 Submitted 2/02; Published 1/03

Latent Dirichlet Allocation

David M. Blei BLEI@CS.BERKELEY.EDU
Computer Science Division
University of California
Berkeley, CA 94720, USA

Andrew Y. Ng ANG@CS.STANFORD.EDU
Computer Science Department
Stanford University
Stanford, CA 94305, USA

Michael I. Jordan JORDAN@CS.BERKELEY.EDU
Computer Science Division and Department of Statistics
University of California
Berkeley, CA 94720, USA

Editor: John Lafferty

Abstract

We describe *latent Dirichlet allocation* (LDA), a generative probabilistic model for collections of discrete data such as text corpora. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, the topic probabilities provide an explicit representation of a document. We present efficient approximate inference techniques based on variational methods and an EM algorithm for empirical Bayes parameter estimation. We report results in document modeling, text classification, and collaborative filtering, comparing to a mixture of unigrams model and the probabilistic LSI model.

1. Introduction

In this paper we consider the problem of modeling text corpora and other collections of discrete data. The goal is to find short descriptions of the members of a collection that enable efficient processing of large collections while preserving the essential statistical relationships that are useful for basic tasks such as classification, novelty detection, summarization, and similarity and relevance judgments.

Significant progress has been made on this problem by researchers in the field of information retrieval (IR) (Baeza-Yates and Ribeiro-Neto, 1999). The basic methodology proposed by IR researchers for text corpora—a methodology successfully deployed in modern Internet search engines—reduces each document in the corpus to a vector of real numbers, each of which represents ratios of counts. In the popular *tf-idf* scheme (Salton and McGill, 1983), a basic vocabulary of "words" or "terms" is chosen, and, for each document in the corpus, a count is formed of the number of occurrences of each word. After suitable normalization, this term frequency count is compared to an inverse document frequency count, which measures the number of occurrences of a word in the entire corpus (generally on a log scale, and again suitably normalized). The end result is a term-by-document matrix X whose columns contain the *tf-idf* values for each of the documents in the corpus. Thus the *tf-idf* scheme reduces documents of arbitrary length to fixed-length lists of numbers.

While the *tf-idf* reduction has some appealing features—notably in its basic identification of sets of words that are discriminative for documents in the collection—the approach also provides a relatively small amount of reduction in description length and reveals little in the way of inter- or intra-document statistical structure. To address these shortcomings, IR researchers have proposed several other dimensionality reduction techniques, most notably *latent semantic indexing (LSI)* (Deerwester et al., 1990). LSI uses a singular value decomposition of the X matrix to identify a linear subspace in the space of *tf-idf* features that captures most of the variance in the collection. This approach can achieve significant compression in large collections. Furthermore, Deerwester et al. argue that the derived features of LSI, which are linear combinations of the original *tf-idf* features, can capture some aspects of basic linguistic notions such as synonymy and polysemy.

To substantiate the claims regarding LSI, and to study its relative strengths and weaknesses, it is useful to develop a generative probabilistic model of text corpora and to study the ability of LSI to recover aspects of the generative model from data (Papadimitriou et al., 1998). Given a generative model of text, however, it is not clear why one should adopt the LSI methodology—one can attempt to proceed more directly, fitting the model to data using maximum likelihood or Bayesian methods.

A significant step forward in this regard was made by Hofmann (1999), who presented the *probabilistic LSI (pLSI)* model, also known as the *aspect model*, as an alternative to LSI. The pLSI approach, which we describe in detail in Section 4.3, models each word in a document as a sample from a mixture model, where the mixture components are multinomial random variables that can be viewed as representations of "topics." Thus each word is generated from a single topic, and different words in a document may be generated from different topics. Each document is represented as a list of mixing proportions for these mixture components and thereby reduced to a probability distribution on a fixed set of topics. This distribution is the "reduced description" associated with the document.

While Hofmann's work is a useful step toward probabilistic modeling of text, it is incomplete in that it provides no probabilistic model at the level of documents. In pLSI, each document is represented as a list of numbers (the mixing proportions for topics), and there is no generative probabilistic model for these numbers. This leads to several problems: (1) the number of parameters in the model grows linearly with the size of the corpus, which leads to serious problems with overfitting, and (2) it is not clear how to assign probability to a document outside of the training set.

To see how to proceed beyond pLSI, let us consider the fundamental probabilistic assumptions underlying the class of dimensionality reduction methods that includes LSI and pLSI. All of these methods are based on the "bag-of-words" assumption—that the order of words in a document can be neglected. In the language of probability theory, this is an assumption of *exchangeability* for the words in a document (Aldous, 1985). Moreover, although less often stated formally, these methods also assume that documents are exchangeable; the specific ordering of the documents in a corpus can also be neglected.

A classic representation theorem due to de Finetti (1990) establishes that any collection of exchangeable random variables has a representation as a mixture distribution—in general an infinite mixture. Thus, if we wish to consider exchangeable representations for documents and words, we need to consider mixture models that capture the exchangeability of both words and documents.

This line of thinking leads to the *latent Dirichlet allocation (LDA)* model that we present in the current paper.

It is important to emphasize that an assumption of exchangeability is not equivalent to an assumption that the random variables are independent and identically distributed. Rather, exchangeability essentially can be interpreted as meaning "*conditionally* independent and identically distributed," where the conditioning is with respect to an underlying latent parameter of a probability distribution. Conditionally, the joint distribution of the random variables is simple and factored while marginally over the latent parameter, the joint distribution can be quite complex. Thus, while an assumption of exchangeability is clearly a major simplifying assumption in the domain of text modeling, and its principal justification is that it leads to methods that are computationally efficient, the exchangeability assumptions do not necessarily lead to methods that are restricted to simple frequency counts or linear operations. We aim to demonstrate in the current paper that, by taking the de Finetti theorem seriously, we can capture significant intra-document statistical structure via the mixing distribution.

It is also worth noting that there are a large number of generalizations of the basic notion of exchangeability, including various forms of partial exchangeability, and that representation theorems are available for these cases as well (Diaconis, 1988). Thus, while the work that we discuss in the current paper focuses on simple "bag-of-words" models, which lead to mixture distributions for single words (unigrams), our methods are also applicable to richer models that involve mixtures for larger structural units such as n-grams or paragraphs.

The paper is organized as follows. In Section 2 we introduce basic notation and terminology. The LDA model is presented in Section 3 and is compared to related latent variable models in Section 4. We discuss inference and parameter estimation for LDA in Section 5. An illustrative example of fitting LDA to data is provided in Section 6. Empirical results in text modeling, text classification and collaborative filtering are presented in Section 7. Finally, Section 8 presents our conclusions.

2. Notation and terminology

We use the language of text collections throughout the paper, referring to entities such as "words," "documents," and "corpora." This is useful in that it helps to guide intuition, particularly when we introduce latent variables which aim to capture abstract notions such as topics. It is important to note, however, that the LDA model is not necessarily tied to text, and has applications to other problems involving collections of data, including data from domains such as collaborative filtering, content-based image retrieval and bioinformatics. Indeed, in Section 7.3, we present experimental results in the collaborative filtering domain.

Formally, we define the following terms:

- A *word* is the basic unit of discrete data, defined to be an item from a vocabulary indexed by $\{1,\ldots,V\}$. We represent words using unit-basis vectors that have a single component equal to one and all other components equal to zero. Thus, using superscripts to denote components, the vth word in the vocabulary is represented by a V-vector w such that $w^v = 1$ and $w^u = 0$ for $u \neq v$.
- A *document* is a sequence of N words denoted by $\mathbf{w} = (w_1, w_2, \ldots, w_N)$, where w_n is the nth word in the sequence.
- A *corpus* is a collection of M documents denoted by $\mathcal{D} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\}$.

We wish 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.

3. Latent Dirichlet allocation

Latent Dirichlet allocation (LDA) is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words.[1]

LDA assumes the following generative process for each document $\mathbf{w}$ in a corpus $\mathcal{D}$:

1. Choose $N \sim \text{Poisson}(\xi)$.
2. Choose $\theta \sim \text{Dir}(\alpha)$.
3. For each of the N words w_n:
   (a) Choose a topic $z_n \sim \text{Multinomial}(\theta)$.
   (b) Choose a word w_n from $p(w_n \mid z_n, \beta)$, a multinomial probability conditioned on the topic z_n.

Several simplifying assumptions are made in this basic model, some of which we remove in subsequent sections. First, the dimensionality k of the Dirichlet distribution (and thus the dimensionality of the topic variable z) is assumed known and fixed. Second, the word probabilities are parameterized by a $k \times V$ matrix β where $\beta_{ij} = p(w^j = 1 \mid z^i = 1)$, which for now we treat as a fixed quantity that is to be estimated. Finally, the Poisson assumption is not critical to anything that follows and more realistic document length distributions can be used as needed. Furthermore, note that N is independent of all the other data generating variables (θ and $\mathbf{z}$). It is thus an ancillary variable and we will generally ignore its randomness in the subsequent development.

A k-dimensional Dirichlet random variable θ can take values in the $(k-1)$-simplex (a k-vector θ lies in the $(k-1)$-simplex if $\theta_i \geq 0$, $\sum_{i=1}^{k} \theta_i = 1$), and has the following probability density on this simplex:

$$p(\theta \mid \alpha) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1}, \tag{1}$$

where the parameter α is a k-vector with components $\alpha_i > 0$, and where $\Gamma(x)$ is the Gamma function. The Dirichlet is a convenient distribution on the simplex — it is in the exponential family, has finite dimensional sufficient statistics, and is conjugate to the multinomial distribution. In Section 5, these properties will facilitate the development of inference and parameter estimation algorithms for LDA.

Given the parameters α and β, the joint distribution of a topic mixture θ, a set of N topics $\mathbf{z}$, and a set of N words $\mathbf{w}$ is given by:

$$p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta) = p(\theta \mid \alpha) \prod_{n=1}^{N} p(z_n \mid \theta) p(w_n \mid z_n, \beta), \tag{2}$$

1. We refer to the latent multinomial variables in the LDA model as topics, so as to exploit text-oriented intuitions, but we make no epistemological claims regarding these latent variables beyond their utility in representing probability distributions on sets of words.

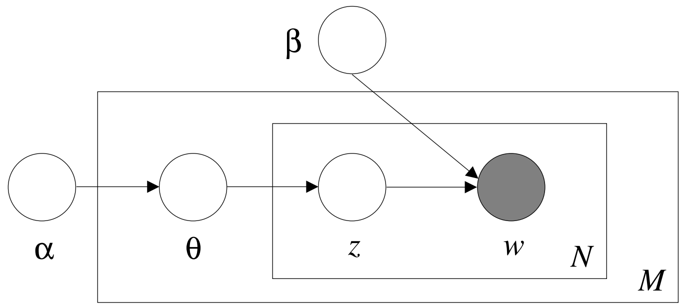


Figure 1: Graphical model representation of LDA. The boxes are "plates" representing replicates. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

where $p(z_n \mid \theta)$ is simply θ_i for the unique i such that $z_n^i = 1$. Integrating over θ and summing over z, we obtain the marginal distribution of a document:

$$p(\mathbf{w} \mid \alpha, \beta) = \int p(\theta \mid \alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n \mid \theta) p(w_n \mid z_n, \beta) \right) d\theta. \tag{3}$$

Finally, taking the product of the marginal probabilities of single documents, we obtain the probability of a corpus:

$$p(\mathcal{D} \mid \alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn} \mid \theta_d) p(w_{dn} \mid z_{dn}, \beta) \right) d\theta_d.$$

The LDA model is represented as a probabilistic graphical model in Figure 1. As the figure makes clear, there are three levels to the LDA representation. The parameters α and β are corpus-level parameters, assumed to be sampled once in the process of generating a corpus. The variables θ_d are document-level variables, sampled once per document. Finally, the variables z_{dn} and w_{dn} are word-level variables and are sampled once for each word in each document.

It is important to distinguish LDA from a simple Dirichlet-multinomial clustering model. A classical clustering model would involve a two-level model in which a Dirichlet is sampled once for a corpus, a multinomial clustering variable is selected once for each document in the corpus, and a set of words are selected for the document conditional on the cluster variable. As with many clustering models, such a model restricts a document to being associated with a single topic. LDA, on the other hand, involves three levels, and notably the topic node is sampled *repeatedly* within the document. Under this model, documents can be associated with multiple topics.

Structures similar to that shown in Figure 1 are often studied in Bayesian statistical modeling, where they are referred to as *hierarchical models* (Gelman et al., 1995), or more precisely as *conditionally independent hierarchical models* (Kass and Steffey, 1989). Such models are also often referred to as *parametric empirical Bayes models*, a term that refers not only to a particular model structure, but also to the methods used for estimating parameters in the model (Morris, 1983). Indeed, as we discuss in Section 5, we adopt the empirical Bayes approach to estimating parameters such as α and β in simple implementations of LDA, but we also consider fuller Bayesian approaches as well.

3.1 LDA and exchangeability

A finite set of random variables $\{z_1, \ldots, z_N\}$ is said to be *exchangeable* if the joint distribution is invariant to permutation. If π is a permutation of the integers from 1 to N:

$$p(z_1, \ldots, z_N) = p(z_{\pi(1)}, \ldots, z_{\pi(N)}).$$

An infinite sequence of random variables is *infinitely exchangeable* if every finite subsequence is exchangeable.

De Finetti's representation theorem states that the joint distribution of an infinitely exchangeable sequence of random variables is as if a random parameter were drawn from some distribution and then the random variables in question were *independent* and *identically distributed*, conditioned on that parameter.

In LDA, we assume that words are generated by topics (by fixed conditional distributions) and that those topics are infinitely exchangeable within a document. By de Finetti's theorem, the probability of a sequence of words and topics must therefore have the form:

$$p(\mathbf{w}, \mathbf{z}) = \int p(\theta) \left(\prod_{n=1}^{N} p(z_n \,|\, \theta) p(w_n \,|\, z_n) \right) d\theta,$$

where θ is the random parameter of a multinomial over topics. We obtain the LDA distribution on documents in Eq. (3) by marginalizing out the topic variables and endowing θ with a Dirichlet distribution.

3.2 A continuous mixture of unigrams

The LDA model shown in Figure 1 is somewhat more elaborate than the two-level models often studied in the classical hierarchical Bayesian literature. By marginalizing over the hidden topic variable z, however, we can understand LDA as a two-level model.

In particular, let us form the word distribution $p(w \,|\, \theta, \beta)$:

$$p(w \,|\, \theta, \beta) = \sum_{z} p(w \,|\, z, \beta) p(z \,|\, \theta).$$

Note that this is a random quantity since it depends on θ.

We now define the following generative process for a document $\mathbf{w}$:

1. Choose $\theta \sim \mathrm{Dir}(\alpha)$.
2. For each of the N words w_n:
   (a) Choose a word w_n from $p(w_n \,|\, \theta, \beta)$.

This process defines the marginal distribution of a document as a continuous mixture distribution:

$$p(\mathbf{w} \,|\, \alpha, \beta) = \int p(\theta \,|\, \alpha) \left(\prod_{n=1}^{N} p(w_n \,|\, \theta, \beta) \right) d\theta,$$

where $p(w_n \,|\, \theta, \beta)$ are the mixture components and $p(\theta \,|\, \alpha)$ are the mixture weights.

Figure 2 illustrates this interpretation of LDA. It depicts the distribution on $p(w \,|\, \theta, \beta)$ which is induced from a particular instance of an LDA model. Note that this distribution on the $(V-1)$-simplex is attained with only $k + kV$ parameters yet exhibits a very interesting multimodal structure.

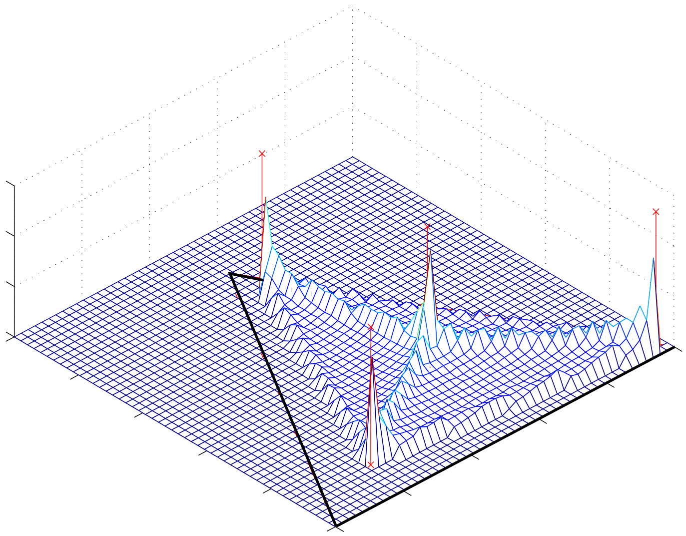


Figure 2: An example density on unigram distributions $p(w \mid \theta, \beta)$ under LDA for three words and four topics. The triangle embedded in the x-y plane is the 2-D simplex representing all possible multinomial distributions over three words. Each of the vertices of the triangle corresponds to a deterministic distribution that assigns probability one to one of the words; the midpoint of an edge gives probability 0.5 to two of the words; and the centroid of the triangle is the uniform distribution over all three words. The four points marked with an x are the locations of the multinomial distributions $p(w \mid z)$ for each of the four topics, and the surface shown on top of the simplex is an example of a density over the $(V-1)$-simplex (multinomial distributions of words) given by LDA.

4. Relationship with other latent variable models

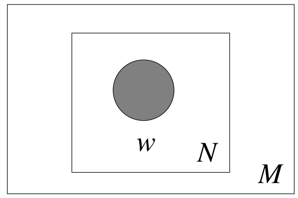
In this section we compare LDA to simpler latent variable models for text—the unigram model, a mixture of unigrams, and the pLSI model. Furthermore, we present a unified geometric interpretation of these models which highlights their key differences and similarities.

4.1 Unigram model

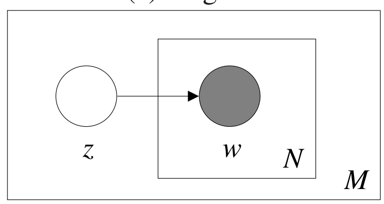
Under the unigram model, the words of every document are drawn independently from a single multinomial distribution:

$$p(\mathbf{w}) = \prod_{n=1}^{N} p(w_n).$$

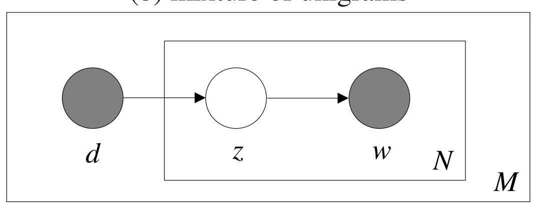
This is illustrated in the graphical model in Figure 3a.



(a) unigram



(b) mixture of unigrams



(c) pLSI/aspect model

Figure 3: Graphical model representation of different models of discrete data.

4.2 Mixture of unigrams

If we augment the unigram model with a discrete random topic variable z (Figure 3b), we obtain a *mixture of unigrams* model (Nigam et al., 2000). Under this mixture model, each document is generated by first choosing a topic z and then generating N words independently from the conditional multinomial $p(w \mid z)$. The probability of a document is:

$$p(\mathbf{w}) = \sum_z p(z) \prod_{n=1}^{N} p(w_n \mid z).$$

When estimated from a corpus, the word distributions can be viewed as representations of topics under the assumption that each document exhibits exactly one topic. As the empirical results in Section 7 illustrate, this assumption is often too limiting to effectively model a large collection of documents.

In contrast, the LDA model allows documents to exhibit multiple topics to different degrees. This is achieved at a cost of just one additional parameter: there are $k-1$ parameters associated with $p(z)$ in the mixture of unigrams, versus the k parameters associated with $p(\theta \mid \alpha)$ in LDA.

4.3 Probabilistic latent semantic indexing

Probabilistic latent semantic indexing (pLSI) is another widely used document model (Hofmann, 1999). The pLSI model, illustrated in Figure 3c, posits that a document label d and a word w_n are conditionally independent given an unobserved topic z:

$$p(d,w_n) = p(d)\sum_z p(w_n \mid z)p(z \mid d).$$

The pLSI model attempts to relax the simplifying assumption made in the mixture of unigrams model that each document is generated from only one topic. In a sense, it does capture the possibility that a document may contain multiple topics since $p(z \mid d)$ serves as the mixture weights of the topics for a particular document d. However, it is important to note that d is a dummy index into the list of documents in the *training set*. Thus, d is a multinomial random variable with as many possible values as there are training documents and the model learns the topic mixtures $p(z \mid d)$ only for those documents on which it is trained. For this reason, pLSI is not a well-defined generative model of documents; there is no natural way to use it to assign probability to a previously unseen document.

A further difficulty with pLSI, which also stems from the use of a distribution indexed by training documents, is that the number of parameters which must be estimated grows linearly with the number of training documents. The parameters for a k-topic pLSI model are k multinomial distributions of size V and M mixtures over the k hidden topics. This gives $kV + kM$ parameters and therefore linear growth in M. The linear growth in parameters suggests that the model is prone to overfitting and, empirically, overfitting is indeed a serious problem (see Section 7.1). In practice, a tempering heuristic is used to smooth the parameters of the model for acceptable predictive performance. It has been shown, however, that overfitting can occur even when tempering is used (Popescul et al., 2001).

LDA overcomes both of these problems by treating the topic mixture weights as a k-parameter hidden *random variable* rather than a large set of individual parameters which are explicitly linked to the training set. As described in Section 3, LDA is a well-defined generative model and generalizes easily to new documents. Furthermore, the $k + kV$ parameters in a k-topic LDA model do not grow with the size of the training corpus. We will see in Section 7.1 that LDA does not suffer from the same overfitting issues as pLSI.

4.4 A geometric interpretation

A good way of illustrating the differences between LDA and the other latent topic models is by considering the geometry of the latent space, and seeing how a document is represented in that geometry under each model.

All four of the models described above—unigram, mixture of unigrams, pLSI, and LDA—operate in the space of distributions over words. Each such distribution can be viewed as a point on the $(V-1)$-simplex, which we call the word simplex.

The unigram model finds a single point on the word simplex and posits that all words in the corpus come from the corresponding distribution. The latent variable models consider k points on the word simplex and form a sub-simplex based on those points, which we call the topic simplex. Note that any point on the topic simplex is also a point on the word simplex. The different latent variable models use the topic simplex in different ways to generate a document.

- The mixture of unigrams model posits that for each document, one of the k points on the word simplex (that is, one of the corners of the topic simplex) is chosen randomly and all the words of the document are drawn from the distribution corresponding to that point.

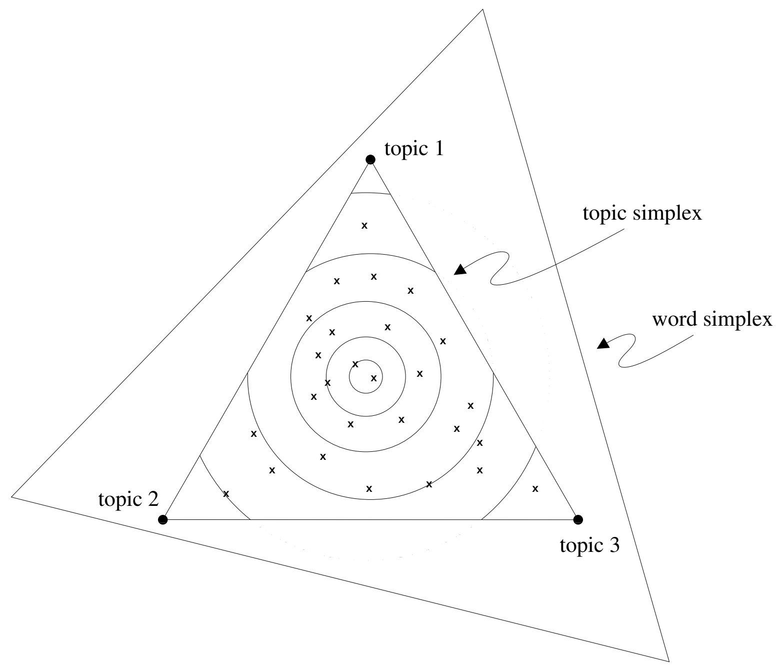


Figure 4: The topic simplex for three topics embedded in the word simplex for three words. The corners of the word simplex correspond to the three distributions where each word (respectively) has probability one. The three points of the topic simplex correspond to three different distributions over words. The mixture of unigrams places each document at one of the corners of the topic simplex. The pLSI model induces an empirical distribution on the topic simplex denoted by x. LDA places a smooth distribution on the topic simplex denoted by the contour lines.

- The pLSI model posits that each word of a *training* document comes from a randomly chosen topic. The topics are themselves drawn from a document-specific distribution over topics, i.e., a point on the topic simplex. There is one such distribution for each document; the set of training documents thus defines an empirical distribution on the topic simplex.

- LDA posits that each word of both the observed and unseen documents is generated by a randomly chosen topic which is drawn from a distribution with a randomly chosen parameter. This parameter is sampled once per document from a smooth distribution on the topic simplex.

These differences are highlighted in Figure 4.

5. Inference and Parameter Estimation

We have described the motivation behind LDA and illustrated its conceptual advantages over other latent topic models. In this section, we turn our attention to procedures for inference and parameter estimation under LDA.

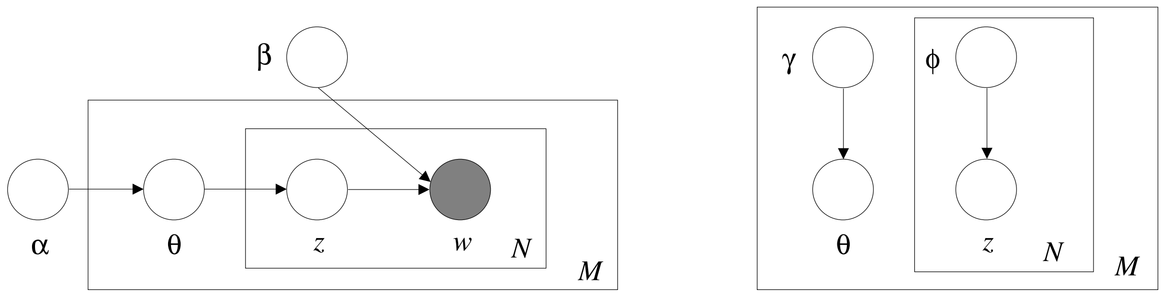


Figure 5: (Left) Graphical model representation of LDA. (Right) Graphical model representation of the variational distribution used to approximate the posterior in LDA.

5.1 Inference

The key inferential problem that we need to solve in order to use LDA is that of computing the posterior distribution of the hidden variables given a document:

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

Unfortunately, this distribution is intractable to compute in general. Indeed, to normalize the distribution we marginalize over the hidden variables and write Eq. (3) in terms of the model parameters:

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

a function which is intractable due to the coupling between θ and β in the summation over latent topics (Dickey, 1983). Dickey shows that this function is an expectation under a particular extension to the Dirichlet distribution which can be represented with special hypergeometric functions. It has been used in a Bayesian context for censored discrete data to represent the posterior on θ which, in that setting, is a random parameter (Dickey et al., 1987).

Although the posterior distribution is intractable for exact inference, a wide variety of approximate inference algorithms can be considered for LDA, including Laplace approximation, variational approximation, and Markov chain Monte Carlo (Jordan, 1999). In this section we describe a simple convexity-based variational algorithm for inference in LDA, and discuss some of the alternatives in Section 8.

5.2 Variational inference

The basic idea of convexity-based variational inference is to make use of Jensen's inequality to obtain an adjustable lower bound on the log likelihood (Jordan et al., 1999). Essentially, one considers a family of lower bounds, indexed by a set of *variational parameters*. The variational parameters are chosen by an optimization procedure that attempts to find the tightest possible lower bound.

A simple way to obtain a tractable family of lower bounds is to consider simple modifications of the original graphical model in which some of the edges and nodes are removed. Consider in particular the LDA model shown in Figure 5 (left). The problematic coupling between θ and β

arises due to the edges between θ, $\mathbf{z}$, and $\mathbf{w}$. By dropping these edges and the $\mathbf{w}$ nodes, and endowing the resulting simplified graphical model with free variational parameters, we obtain a family of distributions on the latent variables. This family is characterized by the following variational distribution:

$$q(\theta, \mathbf{z} \,|\, \gamma, \phi) = q(\theta \,|\, \gamma) \prod_{n=1}^{N} q(z_n \,|\, \phi_n), \tag{4}$$

where the Dirichlet parameter γ and the multinomial parameters $(\phi_1, \ldots, \phi_N)$ are the free variational parameters.

Having specified a simplified family of probability distributions, the next step is to set up an optimization problem that determines the values of the variational parameters γ and ϕ. As we show in Appendix A, the desideratum of finding a tight lower bound on the log likelihood translates directly into the following optimization problem:

$$(\gamma^*, \phi^*) = \arg\min_{(\gamma,\phi)} \mathrm{D}(q(\theta, \mathbf{z} \,|\, \gamma, \phi) \parallel p(\theta, \mathbf{z} \,|\, \mathbf{w}, \alpha, \beta)). \tag{5}$$

Thus the optimizing values of the variational parameters are found by minimizing the Kullback-Leibler (KL) divergence between the variational distribution and the true posterior $p(\theta, \mathbf{z} \,|\, \mathbf{w}, \alpha, \beta)$. This minimization can be achieved via an iterative fixed-point method. In particular, we show in Appendix A.3 that by computing the derivatives of the KL divergence and setting them equal to zero, we obtain the following pair of update equations:

$$\phi_{ni} \propto \beta_{iw_n} \exp\{\mathrm{E}_q[\log(\theta_i) \,|\, \gamma]\} \tag{6}$$

$$\gamma_i = \alpha_i + \textstyle\sum_{n=1}^{N} \phi_{ni}. \tag{7}$$

As we show in Appendix A.1, the expectation in the multinomial update can be computed as follows:

$$\mathrm{E}_q[\log(\theta_i) \,|\, \gamma] = \Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^{k} \gamma_j\right), \tag{8}$$

where Ψ is the first derivative of the $\log\Gamma$ function which is computable via Taylor approximations (Abramowitz and Stegun, 1970).

Eqs. (6) and (7) have an appealing intuitive interpretation. The Dirichlet update is a posterior Dirichlet given expected observations taken under the variational distribution, $\mathrm{E}[z_n \,|\, \phi_n]$. The multinomial update is akin to using Bayes' theorem, $p(z_n \,|\, w_n) \propto p(w_n \,|\, z_n) p(z_n)$, where $p(z_n)$ is approximated by the exponential of the expected value of its logarithm under the variational distribution.

It is important to note that the variational distribution is actually a conditional distribution, varying as a function of $\mathbf{w}$. This occurs because the optimization problem in Eq. (5) is conducted for fixed $\mathbf{w}$, and thus yields optimizing parameters (γ^*, ϕ^*) that are a function of $\mathbf{w}$. We can write the resulting variational distribution as $q(\theta, \mathbf{z} \,|\, \gamma^*(\mathbf{w}), \phi^*(\mathbf{w}))$, where we have made the dependence on $\mathbf{w}$ explicit. Thus the variational distribution can be viewed as an approximation to the posterior distribution $p(\theta, \mathbf{z} \,|\, \mathbf{w}, \alpha, \beta)$.

In the language of text, the optimizing parameters $(\gamma^*(\mathbf{w}), \phi^*(\mathbf{w}))$ are document-specific. In particular, we view the Dirichlet parameters $\gamma^*(\mathbf{w})$ as providing a representation of a document in the topic simplex.

initialize $\phi_{ni}^0 := 1/k$ for all i and n
initialize $\gamma_i := \alpha_i + N/k$ for all i
repeat
    for $n = 1$ **to** N
        for $i = 1$ **to** k
            $\phi_{ni}^{t+1} := \beta_{iw_n} \exp(\Psi(\gamma_i^t))$
        normalize ϕ_n^{t+1} to sum to 1.
    $\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \phi_n^{t+1}$
until convergence

Figure 6: A variational inference algorithm for LDA.

We summarize the variational inference procedure in Figure 6, with appropriate starting points for γ and ϕ_n. From the pseudocode it is clear that each iteration of variational inference for LDA requires $\mathrm{O}((N+1)k)$ operations. Empirically, we find that the number of iterations required for a single document is on the order of the number of words in the document. This yields a total number of operations roughly on the order of N^2k.

5.3 Parameter estimation

In this section we present an empirical Bayes method for parameter estimation in the LDA model (see Section 5.4 for a fuller Bayesian approach). In particular, given a corpus of documents $\mathcal{D} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\}$, we wish to find parameters α and β that maximize the (marginal) log likelihood of the data:

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

As we have described above, the quantity $p(\mathbf{w} \,|\, \alpha, \beta)$ cannot be computed tractably. However, variational inference provides us with a tractable lower bound on the log likelihood, a bound which we can maximize with respect to α and β. We can thus find approximate empirical Bayes estimates for the LDA model via an alternating *variational EM* procedure that maximizes a lower bound with respect to the variational parameters γ and ϕ, and then, for fixed values of the variational parameters, maximizes the lower bound with respect to the model parameters α and β.

We provide a detailed derivation of the variational EM algorithm for LDA in Appendix A.4. The derivation yields the following iterative algorithm:

1. (E-step) For each document, find the optimizing values of the variational parameters $\{\gamma_d^*, \phi_d^* : d \in \mathcal{D}\}$. This is done as described in the previous section.

2. (M-step) Maximize the resulting lower bound on the log likelihood with respect to the model parameters α and β. This corresponds to finding maximum likelihood estimates with expected sufficient statistics for each document under the approximate posterior which is computed in the E-step.

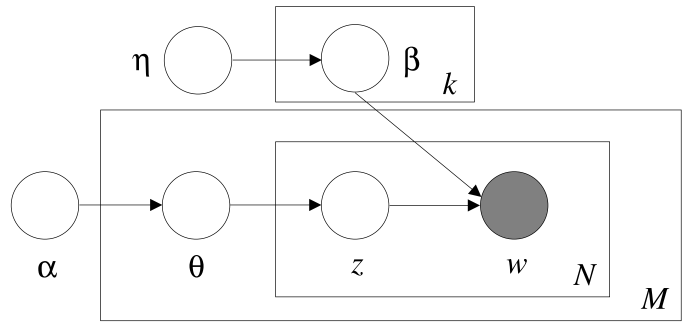


Figure 7: Graphical model representation of the smoothed LDA model.

These two steps are repeated until the lower bound on the log likelihood converges.

In Appendix A.4, we show that the M-step update for the conditional multinomial parameter β can be written out analytically:

$$\beta_{ij} \propto \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi^*_{dni} w^j_{dn}. \tag{9}$$

We further show that the M-step update for Dirichlet parameter α can be implemented using an efficient Newton-Raphson method in which the Hessian is inverted in linear time.

5.4 Smoothing

The large vocabulary size that is characteristic of many document corpora creates serious problems of sparsity. A new document is very likely to contain words that did not appear in any of the documents in a training corpus. Maximum likelihood estimates of the multinomial parameters assign zero probability to such words, and thus zero probability to new documents. The standard approach to coping with this problem is to “smooth” the multinomial parameters, assigning positive probability to all vocabulary items whether or not they are observed in the training set (Jelinek, 1997). Laplace smoothing is commonly used; this essentially yields the mean of the posterior distribution under a uniform Dirichlet prior on the multinomial parameters.

Unfortunately, in the mixture model setting, simple Laplace smoothing is no longer justified as a maximum a posteriori method (although it is often implemented in practice; cf. Nigam et al., 1999). In fact, by placing a Dirichlet prior on the multinomial parameter we obtain an intractable posterior in the mixture model setting, for much the same reason that one obtains an intractable posterior in the basic LDA model. Our proposed solution to this problem is to simply apply variational inference methods to the extended model that includes Dirichlet smoothing on the multinomial parameter.

In the LDA setting, we obtain the extended graphical model shown in Figure 7. We treat β as a $k \times V$ random matrix (one row for each mixture component), where we assume that each row is independently drawn from an exchangeable Dirichlet distribution.[2] We now extend our inference procedures to treat the β_i as random variables that are endowed with a posterior distribution,

2. An exchangeable Dirichlet is simply a Dirichlet distribution with a single scalar parameter η. The density is the same as a Dirichlet (Eq. 1) where $\alpha_i = \eta$ for each component.

conditioned on the data. Thus we move beyond the empirical Bayes procedure of Section 5.3 and consider a fuller Bayesian approach to LDA.

We consider a variational approach to Bayesian inference that places a separable distribution on the random variables β, θ, and $\mathbf{z}$ (Attias, 2000):

$$q(\beta_{1:k}, \mathbf{z}_{1:M}, \theta_{1:M} \,|\, \lambda, \phi, \gamma) = \prod_{i=1}^{k} \mathrm{Dir}(\beta_i \,|\, \lambda_i) \prod_{d=1}^{M} q_d(\theta_d, \mathbf{z}_d \,|\, \phi_d, \gamma_d),$$

where $q_d(\theta, \mathbf{z} \,|\, \phi, \gamma)$ is the variational distribution defined for LDA in Eq. (4). As is easily verified, the resulting variational inference procedure again yields Eqs. (6) and (7) as the update equations for the variational parameters ϕ and γ, respectively, as well as an additional update for the new variational parameter λ:

$$\lambda_{ij} = \eta + \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi^*_{dni} w^j_{dn}.$$

Iterating these equations to convergence yields an approximate posterior distribution on β, θ, and $\mathbf{z}$.

We are now left with the hyperparameter η on the exchangeable Dirichlet, as well as the hyperparameter α from before. Our approach to setting these hyperparameters is again (approximate) empirical Bayes—we use variational EM to find maximum likelihood estimates of these parameters based on the marginal likelihood. These procedures are described in Appendix A.4.

6. Example

In this section, we provide an illustrative example of the use of an LDA model on real data. Our data are 16,000 documents from a subset of the TREC AP corpus (Harman, 1992). After removing a standard list of stop words, we used the EM algorithm described in Section 5.3 to find the Dirichlet and conditional multinomial parameters for a 100-topic LDA model. The top words from some of the resulting multinomial distributions $p(w \,|\, z)$ are illustrated in Figure 8 (top). As we have hoped, these distributions seem to capture some of the underlying topics in the corpus (and we have named them according to these topics).

As we emphasized in Section 4, one of the advantages of LDA over related latent variable models is that it provides well-defined inference procedures for previously unseen documents. Indeed, we can illustrate how LDA works by performing inference on a held-out document and examining the resulting variational posterior parameters.

Figure 8 (bottom) is a document from the TREC AP corpus which was not used for parameter estimation. Using the algorithm in Section 5.1, we computed the variational posterior Dirichlet parameters γ for the article and variational posterior multinomial parameters ϕ_n for each word in the article.

Recall that the ith posterior Dirichlet parameter γ_i is approximately the ith prior Dirichlet parameter α_i plus the expected number of words which were generated by the ith topic (see Eq. 7). Therefore, the prior Dirichlet parameters subtracted from the posterior Dirichlet parameters indicate the expected number of words which were allocated to each topic for a particular document. For the example article in Figure 8 (bottom), most of the γ_i are close to α_i. Four topics, however, are significantly larger (by this, we mean $\gamma_i - \alpha_i \geq 1$). Looking at the corresponding distributions over words identifies the topics which mixed to form this document (Figure 8, top).

Further insight comes from examining the ϕ_n parameters. These distributions approximate $p(z_n \mid \mathbf{w})$ and tend to peak towards one of the k possible topic values. In the article text in Figure 8, the words are color coded according to these values (i.e., the ith color is used if $q_n(z_n^i = 1) > 0.9$). With this illustration, one can identify how the different topics mixed in the document text.

While demonstrating the power of LDA, the posterior analysis also highlights some of its limitations. In particular, the bag-of-words assumption allows words that should be generated by the same topic (e.g., "William Randolph Hearst Foundation") to be allocated to several different topics. Overcoming this limitation would require some form of extension of the basic LDA model; in particular, we might relax the bag-of-words assumption by assuming partial exchangeability or Markovianity of word sequences.

7. Applications and Empirical Results

In this section, we discuss our empirical evaluation of LDA in several problem domains—document modeling, document classification, and collaborative filtering.

In all of the mixture models, the expected complete log likelihood of the data has local maxima at the points where all or some of the mixture components are equal to each other. To avoid these local maxima, it is important to initialize the EM algorithm appropriately. In our experiments, we initialize EM by seeding each conditional multinomial distribution with five documents, reducing their effective total length to two words, and smoothing across the whole vocabulary. This is essentially an approximation to the scheme described in Heckerman and Meila (2001).

7.1 Document modeling

We trained a number of latent variable models, including LDA, on two text corpora to compare the generalization performance of these models. The documents in the corpora are treated as unlabeled; thus, our goal is density estimation—we wish to achieve high likelihood on a held-out test set. In particular, we computed the *perplexity* of a held-out test set to evaluate the models. The perplexity, used by convention in language modeling, is monotonically decreasing in the likelihood of the test data, and is algebraicly equivalent to the inverse of the geometric mean per-word likelihood. A lower perplexity score indicates better generalization performance.[3] More formally, for a test set of M documents, the perplexity is:

$$perplexity(\mathcal{D}_{\text{test}}) = \exp\left\{-\frac{\sum_{d=1}^{M} \log p(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d}\right\}.$$

In our experiments, we used a corpus of scientific abstracts from the C. Elegans community (Avery, 2002) containing 5,225 abstracts with 28,414 unique terms, and a subset of the TREC AP corpus containing 16,333 newswire articles with 23,075 unique terms. In both cases, we held out 10% of the data for test purposes and trained the models on the remaining 90%. In preprocessing the data,

3. Note that we simply use perplexity as a figure of merit for comparing models. The models that we compare are all unigram ("bag-of-words") models, which—as we have discussed in the Introduction—are of interest in the information retrieval context. We are *not* attempting to do language modeling in this paper—an enterprise that would require us to examine trigram or other higher-order models. We note in passing, however, that extensions of LDA could be considered that involve Dirichlet-multinomial over trigrams instead of unigrams. We leave the exploration of such extensions to language modeling to future work.

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Figure 8: An example article from the AP corpus. Each color codes a different factor from which the word is putatively generated.

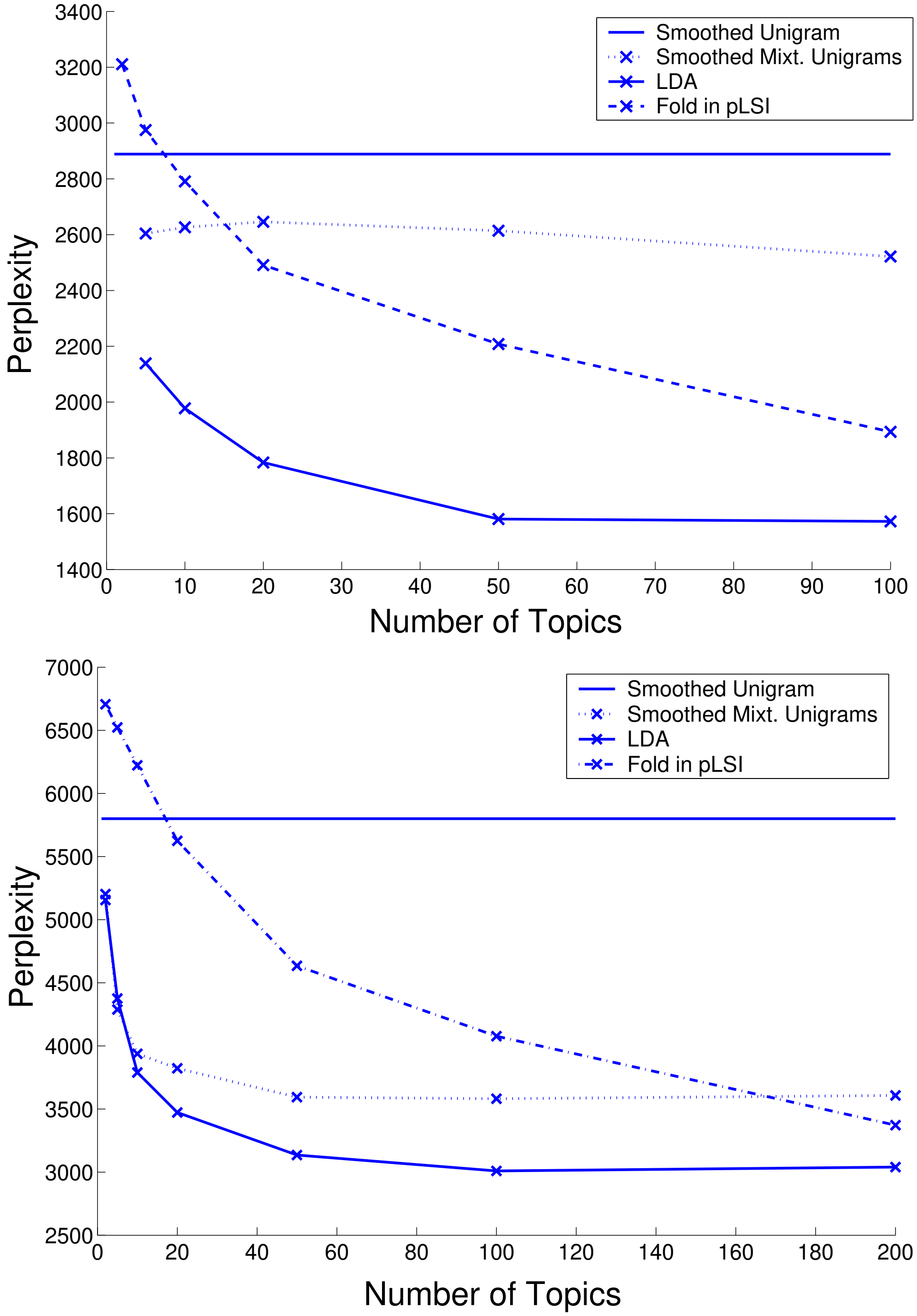


Figure 9: Perplexity results on the nematode (Top) and AP (Bottom) corpora for LDA, the unigram model, mixture of unigrams, and pLSI.

Num. topics (k)	Perplexity (Mult. Mixt.)	Perplexity (pLSI)
2	22,266	7,052
5	2.20×10^{8}	17,588
10	1.93×10^{17}	63,800
20	1.20×10^{22}	2.52×10^{5}
50	4.19×10^{106}	5.04×10^{6}
100	2.39×10^{150}	1.72×10^{7}
200	3.51×10^{264}	1.31×10^{7}

Table 1: Overfitting in the mixture of unigrams and pLSI models for the AP corpus. Similar behavior is observed in the nematode corpus (not reported).

we removed a standard list of 50 stop words from each corpus. From the AP data, we further removed words that occurred only once.

We compared LDA with the unigram, mixture of unigrams, and pLSI models described in Section 4. We trained all the hidden variable models using EM with exactly the same stopping criteria, that the average change in expected log likelihood is less than 0.001%.

Both the pLSI model and the mixture of unigrams suffer from serious overfitting issues, though for different reasons. This phenomenon is illustrated in Table 1. In the mixture of unigrams model, overfitting is a result of peaked posteriors in the training set; a phenomenon familiar in the supervised setting, where this model is known as the naive Bayes model (Rennie, 2001). This leads to a nearly deterministic clustering of the training documents (in the E-step) which is used to determine the word probabilities in each mixture component (in the M-step). A previously unseen document may best fit one of the resulting mixture components, but will probably contain at least one word which did not occur in the training documents that were assigned to that component. Such words will have a very small probability, which causes the perplexity of the new document to explode. As k increases, the documents of the training corpus are partitioned into finer collections and thus induce more words with small probabilities.

In the mixture of unigrams, we can alleviate overfitting through the variational Bayesian smoothing scheme presented in Section 5.4. This ensures that all words will have some probability under every mixture component.

In the pLSI case, the hard clustering problem is alleviated by the fact that each document is allowed to exhibit a different proportion of topics. However, pLSI only refers to the training documents and a different overfitting problem arises that is due to the dimensionality of the $p(z|d)$ parameter. One reasonable approach to assigning probability to a previously unseen document is by marginalizing over d:

$$p(\mathbf{w}) = \sum_{d} \prod_{n=1}^{N} \sum_{z} p(w_n \,|\, z) p(z \,|\, d) p(d).$$

Essentially, we are integrating over the empirical distribution on the topic simplex (see Figure 4).

This method of inference, though theoretically sound, causes the model to overfit. The document-specific topic distribution has some components which are close to zero for those topics that do not appear in the document. Thus, certain words will have very small probability in the estimates of each mixture component. When determining the probability of a new document through marginalization, only those training documents which exhibit a similar proportion of topics will contribute to the likelihood. For a given training document's topic proportions, any word which has small probability in all the constituent topics will cause the perplexity to explode. As k gets larger, the chance that a training document will exhibit topics that cover all the words in the new document decreases and thus the perplexity grows. Note that pLSI does not overfit as quickly (with respect to k) as the mixture of unigrams.

This overfitting problem essentially stems from the restriction that each future document exhibit the same topic proportions as were seen in one or more of the training documents. Given this constraint, we are not free to choose the most likely proportions of topics for the new document. An alternative approach is the "folding-in" heuristic suggested by Hofmann (1999), where one ignores the $p(z|d)$ parameters and refits $p(z|d_{\text{new}})$. Note that this gives the pLSI model an unfair advantage by allowing it to refit $k-1$ parameters to the test data.

LDA suffers from neither of these problems. As in pLSI, each document can exhibit a different proportion of underlying topics. However, LDA can easily assign probability to a new document; no heuristics are needed for a new document to be endowed with a different set of topic proportions than were associated with documents in the training corpus.

Figure 9 presents the perplexity for each model on both corpora for different values of k. The pLSI model and mixture of unigrams are suitably corrected for overfitting. The latent variable models perform better than the simple unigram model. LDA consistently performs better than the other models.

7.2 Document classification

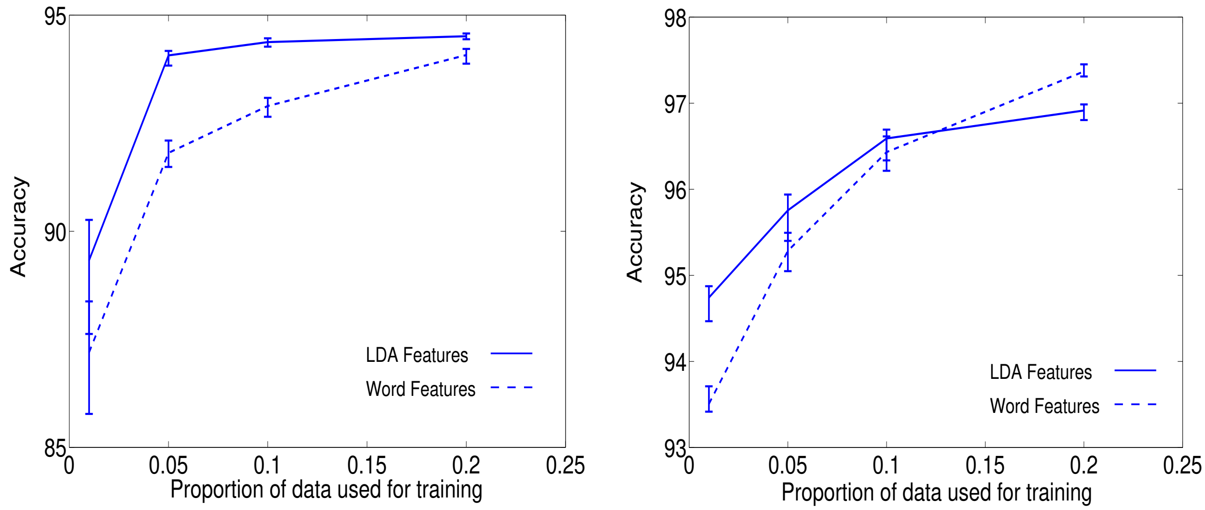
In the text classification problem, we wish to classify a document into two or more mutually exclusive classes. As in any classification problem, we may wish to consider generative approaches or discriminative approaches. In particular, by using one LDA module for each class, we obtain a generative model for classification. It is also of interest to use LDA in the discriminative framework, and this is our focus in this section.

A challenging aspect of the document classification problem is the choice of features. Treating individual words as features yields a rich but very large feature set (Joachims, 1999). One way to reduce this feature set is to use an LDA model for dimensionality reduction. In particular, LDA reduces any document to a fixed set of real-valued features—the posterior Dirichlet parameters $\gamma^*(\mathbf{w})$ associated with the document. It is of interest to see how much discriminatory information we lose in reducing the document description to these parameters.

We conducted two binary classification experiments using the Reuters-21578 dataset. The dataset contains 8000 documents and 15,818 words.

In these experiments, we estimated the parameters of an LDA model on all the documents, without reference to their true class label. We then trained a support vector machine (SVM) on the low-dimensional representations provided by LDA and compared this SVM to an SVM trained on all the word features.

Using the SVMLight software package (Joachims, 1999), we compared an SVM trained on all the word features with those trained on features induced by a 50-topic LDA model. Note that we reduce the feature space by 99.6 percent in this case.



(a) (b)

Figure 10: Classification results on two binary classification problems from the Reuters-21578 dataset for different proportions of training data. Graph (a) is EARN VS. NOT EARN. Graph (b) is GRAIN VS. NOT GRAIN.

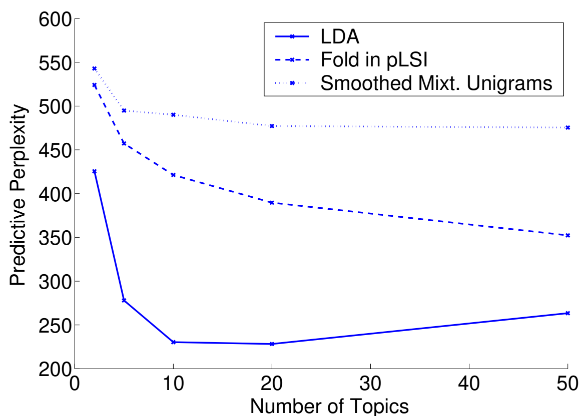


Figure 11: Results for collaborative filtering on the EachMovie data.

Figure 10 shows our results. We see that there is little reduction in classification performance in using the LDA-based features; indeed, in almost all cases the performance is improved with the LDA features. Although these results need further substantiation, they suggest that the topic-based representation provided by LDA may be useful as a fast filtering algorithm for feature selection in text classification.

7.3 Collaborative filtering

Our final experiment uses the EachMovie collaborative filtering data. In this data set, a collection of users indicates their preferred movie choices. A user and the movies chosen are analogous to a document and the words in the document (respectively).

The collaborative filtering task is as follows. We train a model on a fully observed set of users. Then, for each unobserved user, we are shown all but one of the movies preferred by that user and are asked to predict what the held-out movie is. The different algorithms are evaluated according to the likelihood they assign to the held-out movie. More precisely, define the predictive perplexity on M test users to be:

$$\textit{predictive-perplexity}(\mathcal{D}_{\text{test}}) = \exp\left\{-\frac{\sum_{d=1}^{M} \log p(w_{d,N_d} \,|\, \mathbf{w}_{d,1:N_d-1})}{M})\right\}.$$

We restricted the EachMovie dataset to users that positively rated at least 100 movies (a positive rating is at least four out of five stars). We divided this set of users into 3300 training users and 390 testing users.

Under the mixture of unigrams model, the probability of a movie given a set of observed movies is obtained from the posterior distribution over topics:

$$p(w|\mathbf{w}_{\text{obs}}) = \sum_z p(w|z)p(z|\mathbf{w}_{\text{obs}}).$$

In the pLSI model, the probability of a held-out movie is given by the same equation except that $p(z|\mathbf{w}_{\text{obs}})$ is computed by folding in the previously seen movies. Finally, in the LDA model, the probability of a held-out movie is given by integrating over the posterior Dirichlet:

$$p(w|\mathbf{w}_{\text{obs}}) = \int \sum_z p(w|z)p(z|\theta)p(\theta|\mathbf{w}_{\text{obs}})d\theta,$$

where $p(\theta|\mathbf{w}_{\text{obs}})$ is given by the variational inference method described in Section 5.2. Note that this quantity is efficient to compute. We can interchange the sum and integral sign, and compute a linear combination of k Dirichlet expectations.

With a vocabulary of 1600 movies, we find the predictive perplexities illustrated in Figure 11. Again, the mixture of unigrams model and pLSI are corrected for overfitting, but the best predictive perplexities are obtained by the LDA model.

8. Discussion

We have described latent Dirichlet allocation, a flexible generative probabilistic model for collections of discrete data. LDA is based on a simple exchangeability assumption for the words and topics in a document; it is therefore realized by a straightforward application of de Finetti's representation theorem. We can view LDA as a dimensionality reduction technique, in the spirit of LSI, but with proper underlying generative probabilistic semantics that make sense for the type of data that it models.

Exact inference is intractable for LDA, but any of a large suite of approximate inference algorithms can be used for inference and parameter estimation within the LDA framework. We have presented a simple convexity-based variational approach for inference, showing that it yields a fast algorithm resulting in reasonable comparative performance in terms of test set likelihood. Other approaches that might be considered include Laplace approximation, higher-order variational techniques, and Monte Carlo methods. In particular, Leisink and Kappen (2002) have presented a general methodology for converting low-order variational lower bounds into higher-order variational bounds. It is also possible to achieve higher accuracy by dispensing with the requirement of maintaining a bound, and indeed Minka and Lafferty (2002) have shown that improved inferential accuracy can be obtained for the LDA model via a higher-order variational technique known as expectation propagation. Finally, Griffiths and Steyvers (2002) have presented a Markov chain Monte Carlo algorithm for LDA.

LDA is a simple model, and although we view it as a competitor to methods such as LSI and pLSI in the setting of dimensionality reduction for document collections and other discrete corpora, it is also intended to be illustrative of the way in which probabilistic models can be scaled up to provide useful inferential machinery in domains involving multiple levels of structure. Indeed, the principal advantages of generative models such as LDA include their modularity and their extensibility. As a probabilistic module, LDA can be readily embedded in a more complex model—a property that is not possessed by LSI. In recent work we have used pairs of LDA modules to model relationships between images and their corresponding descriptive captions (Blei and Jordan, 2002). Moreover, there are numerous possible extensions of LDA. For example, LDA is readily extended to continuous data or other non-multinomial data. As is the case for other mixture models, including finite mixture models and hidden Markov models, the "emission" probability $p(w_n \mid z_n)$ contributes only a likelihood value to the inference procedures for LDA, and other likelihoods are readily substituted in its place. In particular, it is straightforward to develop a continuous variant of LDA in which Gaussian observables are used in place of multinomials. Another simple extension of LDA comes from allowing mixtures of Dirichlet distributions in the place of the single Dirichlet of LDA. This allows a richer structure in the latent topic space and in particular allows a form of document clustering that is different from the clustering that is achieved via shared topics. Finally, a variety of extensions of LDA can be considered in which the distributions on the topic variables are elaborated. For example, we could arrange the topics in a time series, essentially relaxing the full exchangeability assumption to one of partial exchangeability. We could also consider partially exchangeable models in which we condition on exogenous variables; thus, for example, the topic distribution could be conditioned on features such as "paragraph" or "sentence," providing a more powerful text model that makes use of information obtained from a parser.

Acknowledgements

This work was supported by the National Science Foundation (NSF grant IIS-9988642) and the Multidisciplinary Research Program of the Department of Defense (MURI N00014-00-1-0637). Andrew Y. Ng and David M. Blei were additionally supported by fellowships from the Microsoft Corporation.

References

M. Abramowitz and I. Stegun, editors. *Handbook of Mathematical Functions*. Dover, New York, 1970.

D. Aldous. Exchangeability and related topics. In *École d'été de probabilités de Saint-Flour, XIII—1983*, pages 1–198. Springer, Berlin, 1985.

H. Attias. A variational Bayesian framework for graphical models. In *Advances in Neural Information Processing Systems 12*, 2000.

L. Avery. Caenorrhabditis genetic center bibliography. 2002. URL `http://elegans.swmed.edu/wli/cgcbib`.

R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, New York, 1999.

D. Blei and M. Jordan. Modeling annotated data. Technical Report UCB//CSD-02-1202, U.C. Berkeley Computer Science Division, 2002.

B. de Finetti. *Theory of probability. Vol. 1-2*. John Wiley & Sons Ltd., Chichester, 1990. Reprint of the 1975 translation.

S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

P. Diaconis. Recent progress on de Finetti's notions of exchangeability. In *Bayesian statistics, 3 (Valencia, 1987)*, pages 111–125. Oxford Univ. Press, New York, 1988.

J. Dickey. Multiple hypergeometric functions: Probabilistic interpretations and statistical uses. *Journal of the American Statistical Association*, 78:628–637, 1983.

J. Dickey, J. Jiang, and J. Kadane. Bayesian methods for censored categorical data. *Journal of the American Statistical Association*, 82:773–781, 1987.

A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian data analysis*. Chapman & Hall, London, 1995.

T. Griffiths and M. Steyvers. A probabilistic approach to semantic representation. In *Proceedings of the 24th Annual Conference of the Cognitive Science Society*, 2002.

D. Harman. Overview of the first text retrieval conference (TREC-1). In *Proceedings of the First Text Retrieval Conference (TREC-1)*, pages 1–20, 1992.

D. Heckerman and M. Meila. An experimental comparison of several clustering and initialization methods. *Machine Learning*, 42:9–29, 2001.

T. Hofmann. Probabilistic latent semantic indexing. *Proceedings of the Twenty-Second Annual International SIGIR Conference*, 1999.

F. Jelinek. *Statistical Methods for Speech Recognition*. MIT Press, Cambridge, MA, 1997.

T. Joachims. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*. M.I.T. Press, 1999.

M. Jordan, editor. *Learning in Graphical Models*. MIT Press, Cambridge, MA, 1999.

M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. Introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

R. Kass and D. Steffey. Approximate Bayesian inference in conditionally independent hierarchical models (parametric empirical Bayes models). *Journal of the American Statistical Association*, 84 (407):717–726, 1989.

M. Leisink and H. Kappen. General lower bounds based on computer generated higher order expansions. In *Uncertainty in Artificial Intelligence, Proceedings of the Eighteenth Conference*, 2002.

T. Minka. Estimating a Dirichlet distribution. Technical report, M.I.T., 2000.

T. P. Minka and J. Lafferty. Expectation-propagation for the generative aspect model. In *Uncertainty in Artificial Intelligence (UAI)*, 2002.

C. Morris. Parametric empirical Bayes inference: Theory and applications. *Journal of the American Statistical Association*, 78(381):47–65, 1983. With discussion.

K. Nigam, J. Lafferty, and A. McCallum. Using maximum entropy for text classification. *IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61–67, 1999.

K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134, 2000.

C. Papadimitriou, H. Tamaki, P. Raghavan, and S. Vempala. Latent semantic indexing: A probabilistic analysis. pages 159–168, 1998.

A. Popescul, L. Ungar, D. Pennock, and S. Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Uncertainty in Artificial Intelligence, Proceedings of the Seventeenth Conference*, 2001.

J. Rennie. Improving multi-class text classification with naive Bayes. Technical Report AITR-2001-004, M.I.T., 2001.

G. Ronning. Maximum likelihood estimation of Dirichlet distributions. *Journal of Statistcal Computation and Simulation*, 34(4):215–221, 1989.

G. Salton and M. McGill, editors. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

Appendix A. Inference and parameter estimation

In this appendix, we derive the variational inference procedure (Eqs. 6 and 7) and the parameter maximization procedure for the conditional multinomial (Eq. 9) and for the Dirichlet. We begin by deriving a useful property of the Dirichlet distribution.

A.1 Computing $\mathbf{E}[\log(\theta_i \mid \alpha)]$

The need to compute the expected value of the log of a single probability component under the Dirichlet arises repeatedly in deriving the inference and parameter estimation procedures for LDA. This value can be easily computed from the natural parameterization of the exponential family representation of the Dirichlet distribution.

Recall that a distribution is in the exponential family if it can be written in the form:

$$p(x \mid \eta) = h(x) \exp\left\{\eta^T T(x) - A(\eta)\right\},$$

where η is the natural parameter, $T(x)$ is the sufficient statistic, and $A(\eta)$ is the log of the normalization factor.

We can write the Dirichlet in this form by exponentiating the log of Eq. (1):

$$p(\theta \mid \alpha) = \exp\left\{\left(\sum_{i=1}^k (\alpha_i - 1)\log\theta_i\right) + \log\Gamma\left(\sum_{i=1}^k \alpha_i\right) - \sum_{i=1}^k \log\Gamma(\alpha_i)\right\}.$$

From this form, we immediately see that the natural parameter of the Dirichlet is $\eta_i = \alpha_i - 1$ and the sufficient statistic is $T(\theta_i) = \log\theta_i$. Furthermore, using the general fact that the derivative of the log normalization factor with respect to the natural parameter is equal to the expectation of the sufficient statistic, we obtain:

$$\mathrm{E}[\log\theta_i \mid \alpha] = \Psi(\alpha_i) - \Psi\left(\sum_{j=1}^k \alpha_j\right)$$

where Ψ is the digamma function, the first derivative of the log Gamma function.

A.2 Newton-Raphson methods for a Hessian with special structure

In this section we describe a linear algorithm for the usually cubic Newton-Raphson optimization method. This method is used for maximum likelihood estimation of the Dirichlet distribution (Ronning, 1989, Minka, 2000).

The Newton-Raphson optimization technique finds a stationary point of a function by iterating:

$$\alpha_{\text{new}} = \alpha_{\text{old}} - H(\alpha_{\text{old}})^{-1} g(\alpha_{\text{old}})$$

where $H(\alpha)$ and $g(\alpha)$ are the Hessian matrix and gradient respectively at the point α. In general, this algorithm scales as $\mathrm{O}(N^3)$ due to the matrix inversion.

If the Hessian matrix is of the form:

$$H = \mathrm{diag}(h) + \mathbf{1}z\mathbf{1}^{\mathrm{T}}, \tag{10}$$

where $\mathrm{diag}(h)$ is defined to be a diagonal matrix with the elements of the vector h along the diagonal, then we can apply the matrix inversion lemma and obtain:

$$H^{-1} = \mathrm{diag}(h)^{-1} - \frac{\mathrm{diag}(h)^{-1}\mathbf{1}\mathbf{1}^{\mathrm{T}}\mathrm{diag}(h)^{-1}}{z^{-1} + \sum_{j=1}^k h_j^{-1}}$$

Multiplying by the gradient, we obtain the ith component:

$$(H^{-1}g)_i = \frac{g_i - c}{h_i}$$

where

$$c = \frac{\sum_{j=1}^{k} g_j / h_j}{z^{-1} + \sum_{j=1}^{k} h_j^{-1}}.$$

Observe that this expression depends only on the $2k$ values h_i and g_i and thus yields a Newton-Raphson algorithm that has linear time complexity.

A.3 Variational inference

In this section we derive the variational inference algorithm described in Section 5.1. Recall that this involves using the following *variational distribution*:

$$q(\theta, \mathbf{z} \mid \gamma, \phi) = q(\theta \mid \gamma) \prod_{n=1}^{N} q(z_n \mid \phi_n) \tag{11}$$

as a surrogate for the posterior distribution $p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta)$, where the *variational parameters* γ and ϕ are set via an optimization procedure that we now describe.

Following Jordan et al. (1999), we begin by bounding the log likelihood of a document using Jensen's inequality. Omitting the parameters γ and ϕ for simplicity, we have:

$$\begin{aligned}
\log p(\mathbf{w} \mid \alpha, \beta) &= \log \int \sum_{\mathbf{z}} p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta) d\theta \\
&= \log \int \sum_{\mathbf{z}} \frac{p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta) q(\theta, \mathbf{z})}{q(\theta, \mathbf{z})} d\theta \\
&\geq \int \sum_{\mathbf{z}} q(\theta, \mathbf{z}) \log p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta) d\theta - \int \sum_{\mathbf{z}} q(\theta, \mathbf{z}) \log q(\theta, \mathbf{z}) d\theta \\
&= \mathrm{E}_q[\log p(\theta, \mathbf{z}, \mathbf{w} \mid \alpha, \beta)] - \mathrm{E}_q[\log q(\theta, \mathbf{z})]. \qquad (12)
\end{aligned}$$

Thus we see that Jensen's inequality provides us with a lower bound on the log likelihood for an arbitrary variational distribution $q(\theta, \mathbf{z} \mid \gamma, \phi)$.

It can be easily verified that the difference between the left-hand side and the right-hand side of the Eq. (12) is the KL divergence between the variational posterior probability and the true posterior probability. That is, letting $\mathcal{L}(\gamma, \phi; \alpha, \beta)$ denote the right-hand side of Eq. (12) (where we have restored the dependence on the variational parameters γ and ϕ in our notation), we have:

$$\log p(\mathbf{w} \mid \alpha, \beta) = \mathcal{L}(\gamma, \phi; \alpha, \beta) + \mathrm{D}(q(\theta, \mathbf{z} \mid \gamma, \phi) \parallel p(\theta, \mathbf{z} \mid \mathbf{w}, \alpha, \beta)). \tag{13}$$

This shows that maximizing the lower bound $\mathcal{L}(\gamma, \phi; \alpha, \beta)$ with respect to γ and ϕ is equivalent to minimizing the KL divergence between the variational posterior probability and the true posterior probability, the optimization problem presented earlier in Eq. (5).

We now expand the lower bound by using the factorizations of p and q:

$$\begin{aligned}
\mathcal{L}(\gamma, \phi; \alpha, \beta) = {} & \mathrm{E}_q[\log p(\theta \mid \alpha)] + \mathrm{E}_q[\log p(\mathbf{z} \mid \theta)] + \mathrm{E}_q[\log p(\mathbf{w} \mid \mathbf{z}, \beta)] \\
& - \mathrm{E}_q[\log q(\theta)] - \mathrm{E}_q[\log q(\mathbf{z})].
\end{aligned} \tag{14}$$

Finally, we expand Eq. (14) in terms of the model parameters (α, β) and the variational parameters (γ, ϕ). Each of the five lines below expands one of the five terms in the bound:

$$
\begin{aligned}
L(\gamma, \phi; \alpha, \beta) &= \log \Gamma\left(\textstyle\sum_{j=1}^k \alpha_j\right) - \sum_{i=1}^k \log \Gamma(\alpha_i) + \sum_{i=1}^k (\alpha_i - 1)\left(\Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^k \gamma_j\right)\right) \\
&\quad + \sum_{n=1}^N \sum_{i=1}^k \phi_{ni}\left(\Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^k \gamma_j\right)\right) \\
&\quad + \sum_{n=1}^N \sum_{i=1}^k \sum_{j=1}^V \phi_{ni} w_n^j \log \beta_{ij} \qquad (15) \\
&\quad - \log \Gamma\left(\textstyle\sum_{j=1}^k \gamma_j\right) + \sum_{i=1}^k \log \Gamma(\gamma_i) - \sum_{i=1}^k (\gamma_i - 1)\left(\Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^k \gamma_j\right)\right) \\
&\quad - \sum_{n=1}^N \sum_{i=1}^k \phi_{ni} \log \phi_{ni},
\end{aligned}
$$

where we have made use of Eq. (8).

In the following two sections, we show how to maximize this lower bound with respect to the variational parameters ϕ and γ.

A.3.1 VARIATIONAL MULTINOMIAL

We first maximize Eq. (15) with respect to ϕ_{ni}, the probability that the nth word is generated by latent topic i. Observe that this is a constrained maximization since $\sum_{i=1}^k \phi_{ni} = 1$.

We form the Lagrangian by isolating the terms which contain ϕ_{ni} and adding the appropriate Lagrange multipliers. Let β_{iv} be $p(w_n^v = 1 \mid z^i = 1)$ for the appropriate v. (Recall that each w_n is a vector of size V with exactly one component equal to one; we can select the unique v such that $w_n^v = 1$):

$$
L_{[\phi_{ni}]} = \phi_{ni}\left(\Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^k \gamma_j\right)\right) + \phi_{ni} \log \beta_{iv} - \phi_{ni} \log \phi_{ni} + \lambda_n\left(\textstyle\sum_{j=1}^k \phi_{ni} - 1\right),
$$

where we have dropped the arguments of L for simplicity, and where the subscript ϕ_{ni} denotes that we have retained only those terms in L that are a function of ϕ_{ni}. Taking derivatives with respect to ϕ_{ni}, we obtain:

$$
\frac{\partial L}{\partial \phi_{ni}} = \Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^k \gamma_j\right) + \log \beta_{iv} - \log \phi_{ni} - 1 + \lambda.
$$

Setting this derivative to zero yields the maximizing value of the variational parameter ϕ_{ni} (cf. Eq. 6):

$$
\phi_{ni} \propto \beta_{iv} \exp\left(\Psi(\gamma_i) - \Psi\left(\textstyle\sum_{j=1}^k \gamma_j\right)\right). \qquad (16)
$$

A.3.2 VARIATIONAL DIRICHLET

Next, we maximize Eq. (15) with respect to γ_i, the ith component of the posterior Dirichlet parameter. The terms containing γ_i are:

$$\begin{aligned}\mathcal{L}_{[\gamma]} = & \sum_{i=1}^{k}(\alpha_i - 1)\left(\Psi(\gamma_i) - \Psi\left(\sum_{j=1}^{k}\gamma_j\right)\right) + \sum_{n=1}^{N}\phi_{ni}\left(\Psi(\gamma_i) - \Psi\left(\sum_{j=1}^{k}\gamma_j\right)\right) \\ & - \log\Gamma\left(\sum_{j=1}^{k}\gamma_j\right) + \log\Gamma(\gamma_i) - \sum_{i=1}^{k}(\gamma_i - 1)\left(\Psi(\gamma_i) - \Psi\left(\sum_{j=1}^{k}\gamma_j\right)\right).\end{aligned}$$

This simplifies to:

$$\mathcal{L}_{[\gamma]} = \sum_{i=1}^{k}\left(\Psi(\gamma_i) - \Psi\left(\sum_{j=1}^{k}\gamma_j\right)\right)\left(\alpha_i + \sum_{n=1}^{N}\phi_{ni} - \gamma_i\right) - \log\Gamma\left(\sum_{j=1}^{k}\gamma_j\right) + \log\Gamma(\gamma_i).$$

We take the derivative with respect to γ_i:

$$\frac{\partial \mathcal{L}}{\partial \gamma_i} = \Psi'(\gamma_i)\left(\alpha_i + \sum_{n=1}^{N}\phi_{ni} - \gamma_i\right) - \Psi'\left(\sum_{j=1}^{k}\gamma_j\right)\sum_{j=1}^{k}\left(\alpha_j + \sum_{n=1}^{N}\phi_{nj} - \gamma_j\right).$$

Setting this equation to zero yields a maximum at:

$$\gamma_i = \alpha_i + \sum_{n=1}^{N}\phi_{ni}. \tag{17}$$

Since Eq. (17) depends on the variational multinomial ϕ, full variational inference requires alternating between Eqs. (16) and (17) until the bound converges.

A.4 Parameter estimation

In this final section, we consider the problem of obtaining empirical Bayes estimates of the model parameters α and β. We solve this problem by using the variational lower bound as a surrogate for the (intractable) marginal log likelihood, with the variational parameters ϕ and γ fixed to the values found by variational inference. We then obtain (approximate) empirical Bayes estimates by maximizing this lower bound with respect to the model parameters.

We have thus far considered the log likelihood for a single document. Given our assumption of exchangeability for the documents, the overall log likelihood of a corpus $\mathcal{D} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\}$ is the sum of the log likelihoods for individual documents; moreover, the overall variational lower bound is the sum of the individual variational bounds. In the remainder of this section, we abuse notation by using $\mathcal{L}$ for the total variational bound, indexing the document-specific terms in the individual bounds by d, and summing over all the documents.

Recall from Section 5.3 that our overall approach to finding empirical Bayes estimates is based on a variational EM procedure. In the variational E-step, discussed in Appendix A.3, we maximize the bound $\mathcal{L}(\gamma, \phi; \alpha, \beta)$ with respect to the variational parameters γ and ϕ. In the M-step, which we describe in this section, we maximize the bound with respect to the model parameters α and β. The overall procedure can thus be viewed as coordinate ascent in $\mathcal{L}$.

A.4.1 CONDITIONAL MULTINOMIALS

To maximize with respect to β, we isolate terms and add Lagrange multipliers:

$$\mathcal{L}_{[\beta]} = \sum_{d=1}^{M} \sum_{n=1}^{N_d} \sum_{i=1}^{k} \sum_{j=1}^{V} \phi_{dni} w_{dn}^{j} \log \beta_{ij} + \sum_{i=1}^{k} \lambda_i \left(\sum_{j=1}^{V} \beta_{ij} - 1 \right).$$

We take the derivative with respect to β_{ij}, set it to zero, and find:

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

A.4.2 DIRICHLET

The terms which contain α are:

$$\mathcal{L}_{[\alpha]} = \sum_{d=1}^{M} \left(\log \Gamma \left(\sum_{j=1}^{k} \alpha_j \right) - \sum_{i=1}^{k} \log \Gamma(\alpha_i) + \sum_{i=1}^{k} \left((\alpha_i - 1) \left(\Psi(\gamma_{di}) - \Psi \left(\sum_{j=1}^{k} \gamma_{dj} \right) \right) \right) \right)$$

Taking the derivative with respect to α_i gives:

$$\frac{\partial \mathcal{L}}{\partial \alpha_i} = M \left(\Psi \left(\sum_{j=1}^{k} \alpha_j \right) - \Psi(\alpha_i) \right) + \sum_{d=1}^{M} \left(\Psi(\gamma_{di}) - \Psi \left(\sum_{j=1}^{k} \gamma_{dj} \right) \right)$$

This derivative depends on α_j, where $j \neq i$, and we therefore must use an iterative method to find the maximal α. In particular, the Hessian is in the form found in Eq. (10):

$$\frac{\partial \mathcal{L}}{\partial \alpha_i \alpha_j} = \delta(i,j) M \Psi'(\alpha_i) - \Psi' \left(\sum_{j=1}^{k} \alpha_j \right),$$

and thus we can invoke the linear-time Newton-Raphson algorithm described in Appendix A.2.

Finally, note that we can use the same algorithm to find an empirical Bayes point estimate of η, the scalar parameter for the exchangeable Dirichlet in the smoothed LDA model in Section 5.4.