# Latent Dirichlet Allocation

## Method

### Latent Dirichlet Allocation

#### Notations

* A *word* is the basic unit of the count data of interest in this paper. A word is represented with a -vector, where if and only if is the -th word in the vocabulary shared across the whole dataset.
* A *document* is a sequence of words, denoted by , where is the total number of words in the document and is treated as an ancillary statistic.
* A *corpus* is a set of documents denoted by .

The goal is to find a probabilistic model of a corpus.

#### Latent Dirichlet Allocation

The Latent Dirichlet Allocation (LDA) model for a single document with words can be written in the following hierarchical form:

where is the *topic* of word , is a matrix with being the number of topics, and is the -th row of .

For a corpus with documents where document has words, the LDA model assumes that the documents are independently generated from the above process. Figure 1 shows a graphical model representation of LDA.

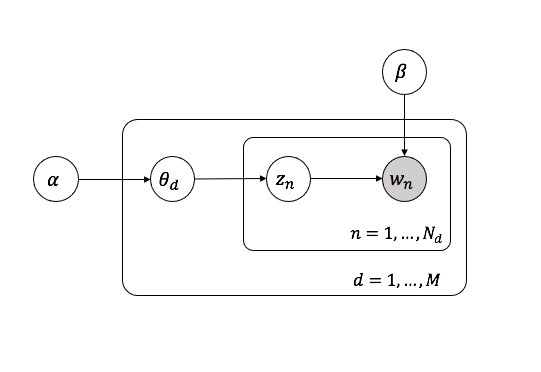


Figure 1. A graphical model representation of LDA

### Algorithm for Inference

#### The Variational EM Algorithm for LDA

In this section, we present a variational EM algorithm to find estimates of parameters that maximizes the marginal log likelihood of the data:

The key challenge is that the marginal density is intractable, which motivates the use of variational inference to obtain a tractable family of lower bounds on the log likelihood. As shown in Figure 2, we eliminate the edges between and drop the nodes, and consider the family characterized by the follwing variational distributions:

where the Dirichlet parameter and the multinomial parameters are the free variational parameters. The optimizing values are found by minimizing the Kullback-Leibler (KL) divergence between the variational distribution and the original posterior . Specifically, for each single document, can be found with an iterative fixed-point method with the following update equations:

With this lower bound obtained from variational inference, we can find approximate empirical Bayes estimates for the LDA model via a variational EM procedure that alternates between the E-step of maximizing the lower bound with respect to the variational parameters and the M-step of maximizing the lower bound with respect to the model parameters for fixed values of . In the M-step, can be solved analytically. The derivations of the updates can be found in cite blei lda.

In the M-step, there is no analytical form of the optimal , so it is updated with Newton-Raphson method. Due to the constraint that the elements of must be positive, we let and solve the unconstrained optimization problem with respect to instead.

#### Algorithm Summary

After initalizing , the variational EM algorithm iterates over the following steps:

* E-step
* For each document, find the optimizing values :
* M-step
  + Update :
  + Update :
  + Iterate the following until convergence:

The derivations of the Newton-Raphson updates are included in the Appendix. Note that if , then the updates associated with the two words are identical. Hence in the implementation, we only store the parameter values for the distinct words to reduce space complexity.

### Implementation

We provide two versions of implementation of LDA, one is coded from scratch (class wrapper1 in our package) while the other is partially adapted from github repo here (class wrapper2 in our package).

(should plain/optimized be an argument of the common method fit\_lda???)

The method fit\_lda in wrapper1 is implemented in plain python, whose performance and diagostics will be presented later. The method fit\_lda\_plain in wrapper2 provides a plain python version of LDA without any optimization, while the fit\_lda\_optimized is optimized with the following techniques:

* Parallelism
* Cython
* ...

### Discussion

* Use MVN for flexible covariance structure of topics/words
* application to classification
* potential identifiability issues

## Appendix

### Newton-Raphson Updates

As shown in cite Blei,

hence