STATS601 final project report
Latent Dirichlet Allocation

Khoa Do

April 2025

1 Introduction

Latent Dirichlet Allocation (LDA) is a Bayesian Topic Model, Mixture of Membership Model, used to model collections of documents and text corpora. In this model, each document is a collection of words, and each word's appearance probability is attributed to the set of topics associated with that document. LDA was proposed by [3] in 2000 in the context of population genetics, and is appied in machine learning by [1] in 2003. Its uses are beyond natural language processing and apply to fields such as Clinical psychology, social science, genetics ...

This report attempts to study LDA and its mathematical implications as presented by [1]. At the time of the paper, progress has been made on modeling text corpora. An example is the *tf-idf* scheme [4], which counts the frequency of words and compares to an inverse document frequency count to determine the significance of words to a document. While the tf-idf reduction gives a small amount of reduction to the feature space, more dimensionality reduction techniques such as *latent semantic indexing (LSI)* address its shortcomings using singular value decomposition. LSI can achieve significant compression in large collections and capture linguistic aspects such as synonymy and polysemy.

In 1999 the paper [2] creates the *probabilistic LSI (pLSI)* model, also known as the *aspect model*. This probabilistic modeling approach models each document as a mixture model, where mixture components are multinomial random variables that can be viewed as "topics". This work, however, does not provide a probabilistic model at the level of documents. To proceed onwards from such results, the authors of [1] present the *Latent Dirichlet Allocation (LDA)* as a Probabilistic Graphical Model that imposes a topic distribution at the document level. The model has became a widely used method in Probabilistic Modeling and is till an active area of research these days.

The report is organized as follows. Section 2 overviews some background mathematical concepts that is helpful toward the rest of the report. We especially look into Variational Inference and its implications for approximating intractable distributions. Section 3 presents the structure of the Latent Dirichlet Allocation, as well as variational methods to do inference and parameter estimation on the model. Section 4 illustrates the empirical results of LDA. We attempt to reproduce some experiment in [1] on simulated data on a smaller scale. Finally, Section 5 gives our conclusions.

2 Background

In this section, we review some concepts and results that is relevant to the study of Latent Dirichlet Allocation. These concepts and results will be used in the following sections (section 3).

2.1 The Dirichlet distribution

Given $k \in \mathbb{N}$ and a k-vector α where $\alpha_i > 0$, the Dirichlet distribution with parameter α is a probability distribution over the standard $(k-1)$-symplex:

$$\Delta^{k-1} = \left\{ (\theta_1, ..., \theta_k) \in \mathbb{R}^k \middle| \theta_i \geq 0, \sum_i \theta_i = 1 \right\}.$$

For a k-vector $\theta \in \Delta^{k-1}$, the density of the distribution is:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{j=1}^k \alpha_j)}{\prod_{j=1}^k \Gamma(\alpha_j)} \prod_{j=1}^k \theta_j^{\alpha_j - 1},$$

meaning that the Dirichlet distribution belongs to the exponential family.

The expectation of the Dirichlet is:

$$\mathbb{E}[\theta_l] = \tilde{\alpha}_l := \frac{\alpha_l}{\sum_j \alpha_j},$$

and the variances and covariances are:

$$\mathrm{Var}[\theta_l] = \frac{\tilde{\alpha}_l(1-\tilde{\alpha}_l)}{1+\sum_j \alpha_j}, \quad \text{and} \quad \mathrm{Cov}[\theta_l, \theta_k] = -\frac{\tilde{\alpha}_l \tilde{\alpha}_k}{1+\sum_j \alpha_j}.$$

The Dirichlet distribution is used to model the mixing proportions between topics for the Latent Dirichlet Allocation model, as its support can be thought of as the probabilities for a k-way categorical distribution. A useful fact to know is that the Dirichlet distribution is the prior conjugate to the categorical (and multinomial) distribution: consider the model:

$$\theta \sim \mathrm{Dir}(\alpha)$$
$$z_i|\theta \sim \mathrm{Cat}(\theta) \text{ for } i \in 1, ..., n$$

Let $n_j = \sum_i \mathbb{1}_{z_i=j}$ counting the number of times that j appears in the generated sequence. Then the posterior distribution is:

$$\theta|z \sim \hat{\alpha},$$

where $\hat{\alpha}_j = \alpha_j + n_j$. The posterior expectation therefore is

$$\mathbb{E}[\theta_l|z, \alpha] = \frac{\alpha_l + n_l}{n + \sum_j \alpha_j}.$$

2.2 Newton-Raphson method for a specially structured Hessian matrix

The Newton-Raphson method is an iterative algorithm that finds the root of a function. It is also used to find the local optima of functions, if we impose the problem to become finding the root of the derivative. In this case, the update rule is:

$$\alpha \leftarrow \alpha - H(\alpha)^{-1} g(\alpha),$$

where H and g being the Hessian matrix and the gradient of the function at α, respectively. Often times, due to having to inverse an $n \times n$ matrix, where n is the dimension, this algorithm has complexity $O(n^3)$.

Due to reasons which will become clear in following sections (section 3.3), we specifically investigate the following case of the Hessian matrix:

$$H(\alpha) = \mathrm{diag}(h) + \mathbb{1} z \mathbb{1}^\top,$$

where h is a vector and z is a scalar. Using Woodbury's formula,

$$H^{-1} = \mathrm{diag}(h)^{-1} + \frac{\mathrm{diag}(h)^{-1}\mathbb{1}\mathbb{1}^\top \mathrm{diag}(h)^{-1}}{z^{-1} + \sum_i h_i^{-1}} = \mathrm{diag}(h)^{-1} + \frac{(1/h)(1/h)^\top}{z^{-1} + \sum_i h_i^{-1}}$$

Multiplying with the gradient:

$$H^{-1} g = \mathrm{diag}(h)^{-1} g + \frac{(1/h)(1/h)^\top g}{z^{-1} + \sum_i h_i^{-1}} = [(1/h) \cdot g] + c(1/h) = (g - c) \cdot (1/h),$$

In which $c = \frac{(1/h)^\top g}{z^{-1} + \sum_i h_i^{-1}}$ is a constant calculated in linear time, and we use $\cdot$ to denote entry-by-entry multiplication between vectors (giving another vector of similar dimensionality).

Thus, this gives us the following linear Newton-Raphson update rule, instead of a cubic one:

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

The linear-time algorithm looks as follow:

Algorithm 1 Newton Raphson update (α, g, h, z)

▷ *Calculate c in linear time* ◁
$c := \frac{\sum_i g_i/h_i}{z^{-1}+\sum_i h_i^{-1}}$

▷ *Update α in linear time* ◁
for $n = 1$ to N **do**
  $\alpha_i \leftarrow \alpha_i - \frac{g_i - c}{h_i}$

2.3 A first look of Variation Inference

2.3.1 Variational Inference.

When the posterior distribution of hidden variables given data is hard to obtain, an alternative to MCMC methods is to approximate this distribution using Variational inference (VI). Specifically, let z be the hidden variables and x be the data, VI uses a (parameterized) family of distributions $q(z;\nu)$ to approximate the posterior distribution $p(z|x)$. One way to find a "fitted" value of the parameter ν is to minimize the KL divergence:

$$\begin{aligned}\mathrm{KL}(q(z;\nu)||p(z|x)) &= \mathbb{E}_q\left[\frac{\log q(z;v)}{\log p(z|x)}\right] \\ &= \mathbb{E}_q[\log q(z;v)] - \mathbb{E}_q[\log p(z,x) - \log p(x)] \\ &= \mathbb{E}_q[\log q(z;v)] - \mathbb{E}_q[\log p(z,x)] + \log p(x) = \log p(x) - \mathscr{L},\end{aligned}$$

where $\mathscr{L} = \mathbb{E}_q[\log p(z,x)] - \mathbb{E}_q[\log q(z;\nu)]$ is the *evidence lower bound* (ELBO). The non-negativity of the Kullback-Leibner divergence ensures that $\mathscr{L}$ is a lower bound of the evidence $\log p(x)$. Thus minimizing the KL divergence is the same as maximizing the ELBO.

2.3.2 Mean-field Variational Inference.

Consider the following mean-field family of distributions:

$$q(z;\nu) = \prod_i q_i(z_i;\nu_i).$$

We delay the choice of the parameterized distributions q_i for a little bit.

Now, the ELBO could be rewritten as

$$\begin{aligned}\mathscr{L} &= \mathbb{E}_q[\log p(z,x)] - \mathbb{E}_q[\log q(z;\nu)] \\ &= \mathbb{E}[\log p(z,x)] - \sum_i \mathbb{E}_{q_i}[\log q_i(z_i;\nu_i)].\end{aligned}$$

We can use coordinate ascent to optimize the ELBO. Indeed, for a coordinate ν_i, holding other coordinates fixed, we need to optimize the amount

$$\begin{aligned}\mathscr{L}_i &= \mathbb{E}[\log p(z,x)] - \mathbb{E}_{q_i}[\log q_i(z_i;\nu_i)] \\ &= \mathbb{E}_{q_i}\left[\mathbb{E}_{q_{-i}}[\log p(z_{-i},z_i,x)]\right] - \mathbb{E}_{q_i}[\log q_i(z_i;\nu_i)] \\ &= \mathbb{E}_{q_i}[\log g(z_i,\nu_{-i},x)] - \mathbb{E}_{q_i}[\log q_i(z_i;\nu_i)],\end{aligned}$$

where $g(z_i,\nu_{-i},x) = \exp\left(\mathbb{E}_{q_{-i}}[\log p(Z_{-i},z_i,x)]\right)$ is a function of z_i.

We know that the maximum is achieved when

$$g(z_i,\nu_{-i},x) \propto q_i(z_i;\nu_i) \Leftrightarrow \exp\left(\mathbb{E}_{q_{-i}}[\log p(z_{-i},z_i,x)]\right) \propto q_i(z_i;\nu_i) \tag{1}$$

At this point, we are not sure if the chosen family q_i is "rich" enough for such a solution ν_i to exist yet! To solve this problem, assume that the following conditional distribution is from the exponential family:

$$p(z_i|z_{-i},x) = \exp(\eta_i(z_{-i},x)^\top g(z_i) - a(\eta_i(z_{-i},x))).$$

This condition would hold for the cases that we will look into. Then,

$$\exp\left(\mathbb{E}_{q_{-i}}[\log p(z_{-i}, z_i, x)]\right) = \exp\left(\mathbb{E}_{q_{-i}}[\eta_i(z_{-i}, x)^\top g(z_i) - a(\eta_i(z_{-i}, x))]\right) \propto \exp(\mathbb{E}_{q_{-i}}[\eta_i(z_{-i}, x)]^\top g(z_i)).$$

Thus, now equation 1 is satisfied if we choose q_i to be in the same exponential family with the same transformation g and the same log normalizer a, then we would have the coordinate update rule

$$\nu_i = \mathbb{E}[\eta_i(z_{-i}, x)].$$

This completes our coordinate ascent algorithm.

2.3.3 VI and the EM algorithm.

We have seen how to approximate a posterior distribution of latent variables using Mean-field VI. Let's see how it can be used in deriving the Variational EM algorithm.

The derivation is similar to the ordinal EM algorithm seen from class. Given data x dependent on the latent variable z, we need to estimate the parameter θ that maximizes the *evidence*, or the log-likelihood

$$\log p(x; \theta).$$

Consider any distribution q on the latent variable. Similar to what we have seen before, the evidence equals KL divergence plus the ELBO:

$$\log p(x; \theta) = \mathrm{KL}(q(z)||p(z|x; \theta)) + \{\mathbb{E}_q[\log p(z, x; \theta)] - \mathbb{E}_q[\log q(z)]\}.$$

Thus, the optimization objective is:

$$\mathrm{KL}(q(z)||p(z|x; \theta)) + \mathbb{E}_q[\log p(z, x; \theta)].$$

Assume the value $\theta = \theta^{(t)}$ at time t, and we need to optimize for a better value $\theta^{(t+1)}$. Then:

1. The ordinal EM algorithm puts $q(z) = p(z|x; \theta^{(t)})$ and optimize $\mathbb{E}_q[\log p(z, x; \theta)]$. This way, at $\theta = \theta^{(t)}$ (before optimization), the KL divergence equals 0, and after optimization (change θ to be $\theta^{(t+1)}$), the KL is non-negative. Therefore, the optimization objective will be non-decreasing.
2. Variational EM first approximates $q(z) = q(z; \nu) \approx p(z|x; \theta^{(t)})$ and also optimize $\mathbb{E}_q[\log p(z, x; \theta)]$. Before optimization, using approximation methods like the one we have seen for the Variational E step, the KL divergence is made as small as possible. Therefore, after optimization, we also would generally expect the optimization objective to increase, although it is not guaranteed.

Variational EM helps out for parameter estimation in the cases that the quantity $\mathbb{E}_q[\log p(z, x; \theta)]$ is intractable or hard to compute for the ordinary choice of q.

3 Latent Dirichlet Allocation

This section studies the theory of Latent Dirichlet Allocation, from its definition to doing posterior sampling, inference, and parameter estimation

3.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a probabilistic model that generates corpus. The generative process for each document $\mathbf{w}$ is as follow:

1. Choose $N \sim \mathrm{Poisson}(\xi)$ representing the total number of words for that document.
2. Choose $\theta \sim \mathrm{Dir}(\alpha)$ representing the topic proportion for that document.
3. For each n of the N words in that document: first choose a topic $z_n \sim \mathrm{Cat}(\theta)$ for that word, and then choose the word $w_n \sim p(\cdot|z_n, \beta)$ following the word distribution from that topic (parameterized by β).

We make some remarks about this model:

1. The dimensionality K of the Dirichlet distribution is assumed known and fixed. Thus $\text{Dir}(\alpha)$ is a distribution over topic vectors of length K.

2. The word probabilities are parameterized by a $k \times V$ matrix β, where $\beta_{ij} = p(w^j = 1|z^i = 1)$ is the probability of the j-th word in topic i (V being the total number of word).

3. The Poisson assumption is not critical, and N is independent of all the other data generating variables. Thus, similar to the paper, we ignore its randomness for subsequent sections.

4. Looking at the generating process, the position and ordering of different words in each document do not play any role in the model. This mean that the model has the "bag-of-words" assumption - it considers each document as a set of word without ordering between them.

5. Each document can be considered as a mixture model, with mixing proportion being random and drawn from a prior distribution. Thus, LDA is a *Mixture of Membership Model*.

3.2 Inference

Assume that the parameters α and β of the model are known to us (we will touch into parameter estimation later). The first problem we want to solve to use LDA is computing the posterior distribution of the hidden variables given a 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 intractable to compute (we refer to Section 5.1 of [1] for more details). Fortunately, a variety of approximate inference algorithms are available. In this section, we present a variational inference approach that is shown in the paper.

To do that, consider the following family of mean-field variational distributions:

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

We aim to use the coordinate ascent algorithm, as presented in Section 2.3, to fit the parameters γ and ϕ.

Mathematical details. Let's work out the mathematics in detail. For a document, the hidden variables are $\{\theta, z_1, z_2, ..., z_N\}$. The corresponding variational parameters are $\{\gamma, \phi_1, \phi_2, ..., \phi_N\}$. As indicated in Section 2.3, we need first to ensure that *the conditional distribution is from the exponential family*. This means verifying two things:

1. Firstly we want to check that the conditional distribution $p(\theta|\mathbf{z}, x)$ is from the exponential family. This is obvious, since this is just a Dirichlet posterior distribution of θ when $\mathbf{z}$ is known. Hence, let $c = \text{Count}(\mathbf{z})$ be a k-vector that counts the appearance of each topics in $\mathbf{z}$, whose entries sum up to N, then we have
$$\theta|\mathbf{z}, \mathbf{w} \sim \theta|\mathbf{z} \sim \text{Dir}(\alpha + c)$$

2. Secondly, without loss of generality, we want to check that the conditional distribution $p(z_i|\mathbf{z}_{-i}, \theta, x)$ is also from the exponential family. Indeed, since z_i is independent of other indexes in $\mathbf{z}_{-i}$, we have
$$z_i = k|\mathbf{z}_{-i}, \theta, \mathbf{w} \sim z_i = k|\theta, w_i \propto \theta_k \beta_{k,w_i} \implies z_i = k|\theta, w_i = C\theta_k \beta_{k,w_i}$$
Where the constant C is dependent on θ (and the parameter β) and will be taken care of shortly. In other words, conditioned on θ, z_i follows a categorical distribution with parameter θ. When w_i is known this becomes a posterior categorical distribution with different weights.

Hence we have confirmed that both of them are from the exponential family. Following what we have derived from Section 2.3, the next step is to *choose q_i's to be in the same exponential families and derive the coordinate update rule*. We have:

1. Choose γ to parameterize a Dirichlet distribution. This means that $\eta(a) = (a - \mathbb{1})$ is a function from $\mathbb{R}^k$ to $\mathbb{R}^k$. Following what we have derived, we want

$$\gamma - \mathbb{1} = \mathbb{E}_\phi[\eta(\alpha + c)] = \mathbb{E}_\phi[\alpha + c - \mathbb{1}] \Leftrightarrow \gamma = \alpha + \mathbb{E}_\phi[c].$$

   But $c = \text{Count}(\mathbf{z})$ counts the number of appearance of each topics in $\mathbf{z}$, hence its expectation is:

$$\mathbb{E}_\phi[c] = \sum_n \phi_n$$

   Thus we derived that $\gamma = \alpha + \sum_n \phi_n$. Note that when $\mathbf{z}$ is known, θ and $\mathbf{w}$ are independent. That is why we don't see the appearance of $\mathbf{w}$ in η.

2. For all n, choose ϕ_n to parameterize a k-way categorical distribution. This is an exponential family distribution, with $\eta : a \in \mathbb{R}^K \mapsto \log(a) \in \mathbb{R}^K$. We want

$$\log(\phi_n) = \mathbb{E}_{\gamma,\phi_{-n}}[\log(C\theta\beta_{\cdot,x_n})].$$

   This means

$$\phi_n = \exp\left(\mathbb{E}_{\gamma,\phi_{-n}}[\log(\theta\beta_{\cdot,x_n})] + \mathbb{E}_{\gamma,\phi_{-n}}[\log(C)]\right) = \exp\left(\mathbb{E}_\gamma[\log(\theta\beta_{\cdot,x_n})] + \mathbb{E}_\gamma[\log(C)]\right)$$

$$\propto \exp\left(\mathbb{E}_\gamma[\log(\theta\beta_{\cdot,x_n})]\right) = \beta_{\cdot,x_n} \exp\left(\mathbb{E}_\gamma[\log\theta]\right).$$

   Thus, we got rid of the constant C as promised! (note that these are equalities between k-vectors, and we can ignore multiplying the entire vector by a constant because we will normalize them later). We also note that ϕ_n is updated independently of other ϕ_m's for $m \neq n$, which is a consequence of their conditional independence following the graphical model.

Finally, to finish the update rules for our coordinate ascent algorithm, notice that we need to know how to calculate the expectation

$$\mathbb{E}_\gamma[\log(\theta_i)]$$

for all $i = 1, 2, ..., k$, where γ parameterizes the Dirichlet distribution of θ. For this, let $g(\gamma) = \frac{\Gamma(\sum_j \gamma_j)}{\prod_j \Gamma(\gamma_j)}$, then recall the pdf of a Dirichlet distribution:

$$g(\gamma) \int_\theta \prod_j \theta_j^{\gamma_j - 1} d\theta = 1$$

Fix an index i. Taking derivative with respect to θ_i yields

$$\begin{aligned}
&\partial_i g \int_\theta \prod_j \theta_j^{\gamma_j - 1} d\theta + g \int_\theta \log\theta_i \prod_j \theta_j^{\gamma_j - 1} d\theta = 0 \\
\implies &\frac{\partial_i g}{g} + \mathbb{E}_\gamma[\log\theta_i] = 0 \\
\implies &\mathbb{E}_\gamma[\log\theta_i] = -\partial_i \log g = \partial_i \left(\log\Gamma(\gamma_i) - \log\Gamma\left(\sum_j \gamma_j\right)\right) \\
\implies &\mathbb{E}_\gamma[\log\theta_i] = \Psi(\gamma_i) - \Psi\left(\sum_i \gamma_i\right),
\end{aligned}$$

where Φ is the derivative of the log of the Gamma function. That is all the math we need to derive the following algorithm:

Algorithm 2 Variational Inference($\mathbf{w}, \alpha, \beta$)

▷ *Initialization* ◁
Initialize $\gamma_i := \alpha_i + N/K$ for all $i \leq K$
Initialize $\phi_{ni} = 1/k$ for all $i \leq k, n \leq N$

▷ *Coordinate ascent loop* ◁
while not converge: **do**
  for $n = 1$ to N **do**
    for $i = 1$ to k **do**
      $\phi_{ni} = \beta_{iw_n} \exp(\Psi(\gamma_i))$
    normalize ϕ_n to sum to 1.
  $\gamma = \alpha + \sum_n \phi_n$
return γ and ϕ_n's

3.3 Parameter estimation

Similar to the previous subsection, the following *evidence* is not tractable:

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

where the sum is taken over all documents $\mathbf{w}_d$. As for that subsection, we use the Variational EM algorithm as a variational alternative.

The E-step of the Variational EM algorithm is to approximate the posterior distribution using Variational Inference. This follows directly from algorithm 2 from section 3.2. Thus, it is left for us to derive the M-step, which is maximizing:

$$L = \sum_{d=1}^{M} \mathbb{E}_{q^d}[\log p(\mathbf{w}^d, \mathbf{z}^d, \theta^d | \alpha, \beta)],$$

where the sum is taken across documents.

We focus on one document:

$$E = \mathbb{E}_q[\log p(\mathbf{w}, \mathbf{z}, \theta | \alpha, \beta)].$$

We have:

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

Taking derivative with respect to α gives:

$$\frac{\partial}{\partial \alpha} E = \mathbb{E}_q \left(\frac{\partial}{\partial \alpha} \log p(\theta | \alpha) \right) = \Psi(|\alpha|) - \Psi(\alpha) - \Psi(|\gamma|) + \Psi(\gamma). \tag{2}$$

Notice that:

1. We use the notation $|\alpha| = \sum_i \alpha_i$ for convenience (the same for other vectors)
2. $\Psi(\alpha)$ and $\Psi(\gamma)$ can be thought of as vectors created from acting Ψ on all entries of α and γ, respectively. $\Psi(|\alpha|)$ and $\Psi(|\gamma|)$ can be thought of as vectors of identical entries.
3. We use a fact that we have proven in section 3.2:

$$\mathbb{E}_\gamma[\log \theta] = \Psi(\gamma) - \Psi(|\gamma|).$$

Next, taking derivative with respect to β gives:

$$\frac{\partial}{\partial \beta} E = \mathbb{E}_q \left(\frac{\partial}{\partial \beta} \sum_n \log p(w_n | z_n, \beta) \right) = \mathbb{E}_q \left(\frac{\partial}{\partial \beta} \sum_n \log \beta_{z_n w_n} \right) = \sum_n \sum_i \frac{\phi_{ni}}{\beta_{iw_n}} \tag{3}$$

Puting these pieces together:

1. **Optimizing β.** The constraint on β is the sum of entries per topic equals 1. Summing equation 3 across documents and adding the derivative of the Lagrange multipliers give us the minimization rule:

$$\beta_{ij} \propto \sum_{d=1}^{M} \sum_{n=1}^{N_d} \theta_{ni}^d w_{dn} \mathbb{1}_{w_{dn}=j}$$

2. **Optimizing** α is more complicated. Using equation 2, the derivative of α across documents is:

$$\frac{\partial}{\partial \alpha} L = M \left(\Psi(|\alpha|) - \Psi(\alpha) \right) - \sum_{d=1}^{M} \left(\Psi(|\gamma|) - \Psi(\gamma) \right)$$

The Hessian matrix is:

$$\frac{\partial}{\partial \alpha_i \alpha_j} L = M \left(\Psi'(|\alpha|) - \mathbb{1}_{i=j} \Psi'(\alpha_i) \right).$$

All in all, we just derived the following algorithm:

Algorithm 3 Parameter estimation $\left((\mathbf{w}^d)_{d=1}^M \right)$

▷ *Initialization* ◁
Save number of topics K and number of vocab P
Save number of documents M and number of words per document N_d's
Initialize $\alpha \in \mathbb{R}^K$: $\alpha_i \leftarrow \frac{1}{K}$.
Initialize $\beta \in \mathbb{R}^{K \times P}$: $\beta_{ij} \leftarrow \frac{1}{P}$.

▷ *Variational EM main loop* ◁
while not converge **do**

  ▷ *E step* ◁
  for Each document $\mathbf{w}^d$: **do**
    $\gamma^d, \phi^d \leftarrow$ Alg.2$(\mathbf{w}^d, \alpha, \beta)$

  ▷ *M step for* α ◁
  while not converge **do**
    $g \leftarrow M \left(\Psi(|\alpha|) - \Psi(\alpha) \right) - \sum_{d=1}^{M} \left(\Psi(|\gamma^d|) - \Psi(\gamma^d) \right)$
    $h \leftarrow -M \Psi'(\alpha) \in \mathbb{R}^K$
    $z \leftarrow M \Psi'(|\alpha|)$
    Update α using Alg.1(α, g, h, z).

  ▷ *M step for* β ◁
  for $i = 1$ to K **do**
    for $j = 1$ to P **do**
      $\beta_{ij} \leftarrow \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi_{ni}^d \mathbb{1}_{w_{dn}=j}$
    Normalize $\beta_{i,\cdot}$ to sum to 1.
return α and β

4 Numerical experiments

In this section, we attempt some numerical experiments on simulated data.

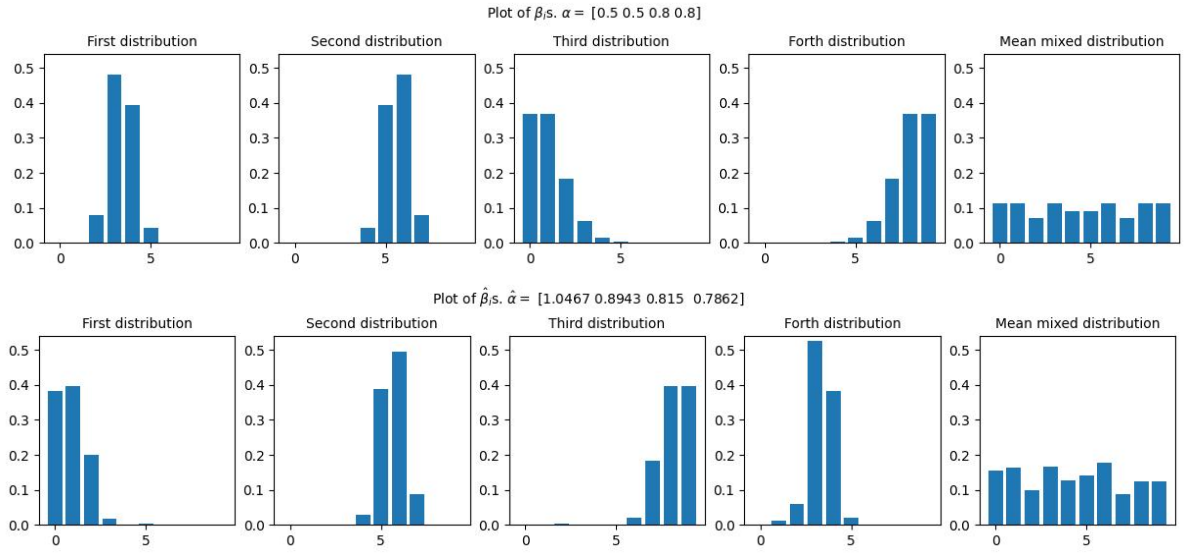

Figure 1: Parameter estimation from well-specified data. First row: Ground truth parameters for one experiment that we ran, with 4 topics. The distribution are discretized from Gaussian and Poisson density, and the mean mixed distribution between them is also shown. Second row: Fitted parameters after 50 Variational EM iterations. The variational algorithm does well in estimating the parameters overall, but the numbers of $\hat{\alpha}$ are slightly higher than the ground truth. Notice that no spacial information between the words is contained in the model; the numbers are just for labeling and the densities are chosen just for ease of interpretation.

4.1 Parameter estimation

We test the effectiveness of the variational approach on estimating the parameters of an LDA model on simulated data. Due to time complexity reason, we simulated data using a dictionary of $P = 10$ words and K from 3 to 4 topics. The discrete distributions for the ground truth of β is chosen by normalizing the density values of the Normal distribution and Poisson distribution at integer lattice points.

An example result of the experiment is shown in 1. The algorithm does a decent job in estimating the parameters of β and the proportion in α (note that the mean mixed distributions are close to each other). However, the numbers in the Dirichlet parameter α are over-estimated in $\hat{\alpha}$. This is a behavior often seen in fitting LDA with variational methods: the model would likely put more mass in the "inner", central area in the simplex of the Dirichlet distribution and make the estimation for the Dirichlet parameter higher.

4.2 Perplexity as a metric of generalization

Following from [1], we test the viability of *perplexity* as a metric of model generalization on simulated data. The perplexity on a hold-out dataset (one that is not used in training and is unseen by the model) is defined as:

$$perplexity(D_{\text{test}}) = \exp \frac{-\sum_{d=1}^{M} \log p(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d}$$

The perplexity is the exponential of the negative the geometric-mean per-word likelihood. If the model selects all the word uniformly as random, then the perplexity of any dataset would be P, the number of word in the dictionary. A smaller perplexity on the hold-out data would indicate better generalization of the model, as higher likelihood is assign to this data.

The perplexity of an LDA model does not have a closed form formula. As we would need to integrate over a Dirichlet distribution. In our study, we used a Monte Carlo approach and sample from the K-way Dirichlet distribution a number of times, and then take the average. Due to the number of topic K being low, the variance of this approach is not too high, as we can see that the plot is reasonably smooth-looking.

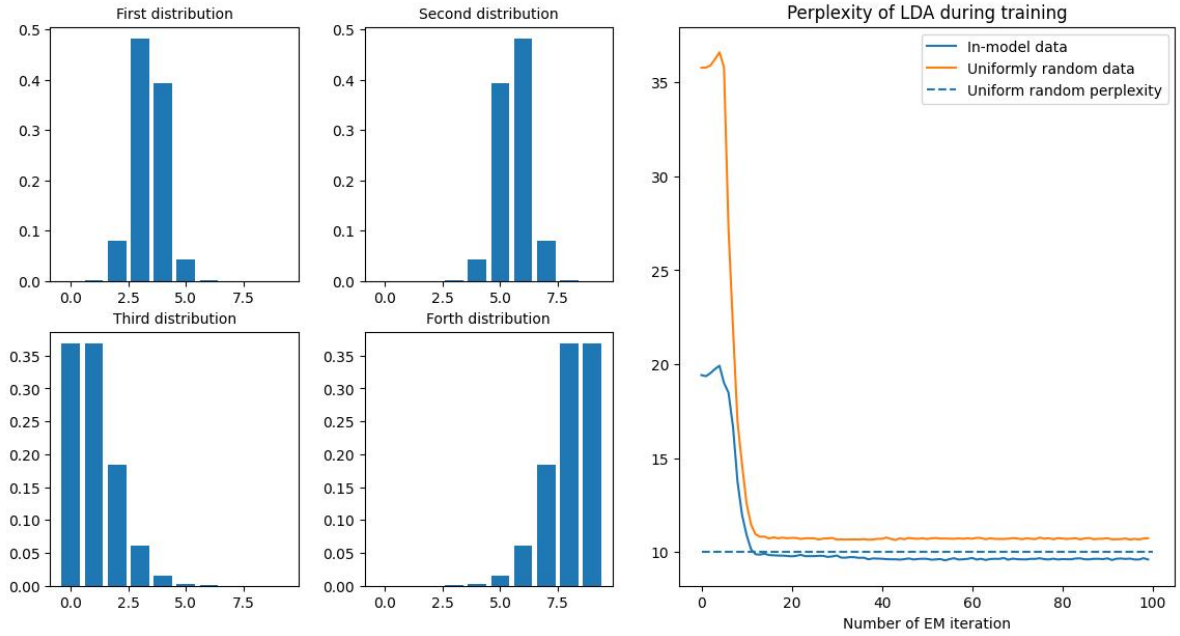

Figure 2: An experiment of measuring the perplexity as a function of number of variational EM iteration. The parameters are the same with the experiment shown in 1. Left: ground truth parameters ($\alpha = [1, 1, 1.5, 1.5]$). Rights: Plot of the perplexity. The different between the lines are relatively small compared to their magnitude.

Figure 2 shows the perplexity of an LDA model during training, under the data generated by the same LDA model in the example shown in Subsection 4.1. The x-axis represents the number of Variational EM iteration. We measure the perplexity on 2 unseen simulated dataset: dataset 1 is generated by the ground truth model (shown in blue), and dataset 2 is a model where words are sampled uniformly as random (shown in orange). We can see that at convergence, the perplexity of dataset 1 drops to lower than the value $P = 10$, while the perplexity of the dataset 2 (uniformly randomly generated and is hence not from the model) stays a little above this value. This behavior is expected from our model, although the margin of differences between the lines are relatively small.

To see the difference in perplexity more viably, we repeat the experiment again under a "shaper", sparser parameter β: each topic assigns high values for 1 or 2 words and very small values (1 percent) to all other words. When testing the LDA on this model, the different in perplexity becomes much more visible (fig 3).

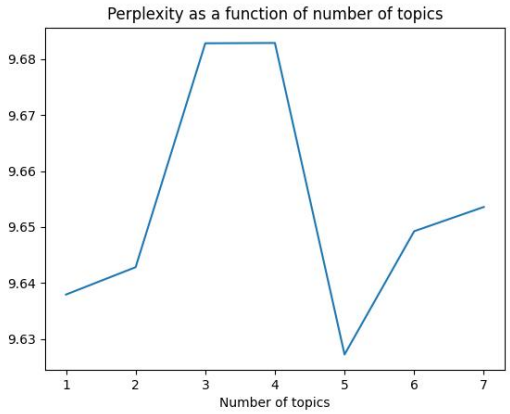

Figure 4: Perplexity as a metric for number-of-topic selection. The parameters are similar to figures 1 and 2. With simulated data, we could not reproduce the authors' result on real data: the number of topics does not effect the perplexity in any clear manner.

Another way to use perplexity is to use it as a metric of selecting the number of topics. In fig 4, we plot the perplexity of an in-model simulated dataset as a function of the number of topics used during parameter estimation. For this experiment, we observed little impact that the number of topics has over the hold-out perplexity. This has less significance compared to what the authors of [1] have observed from experimenting

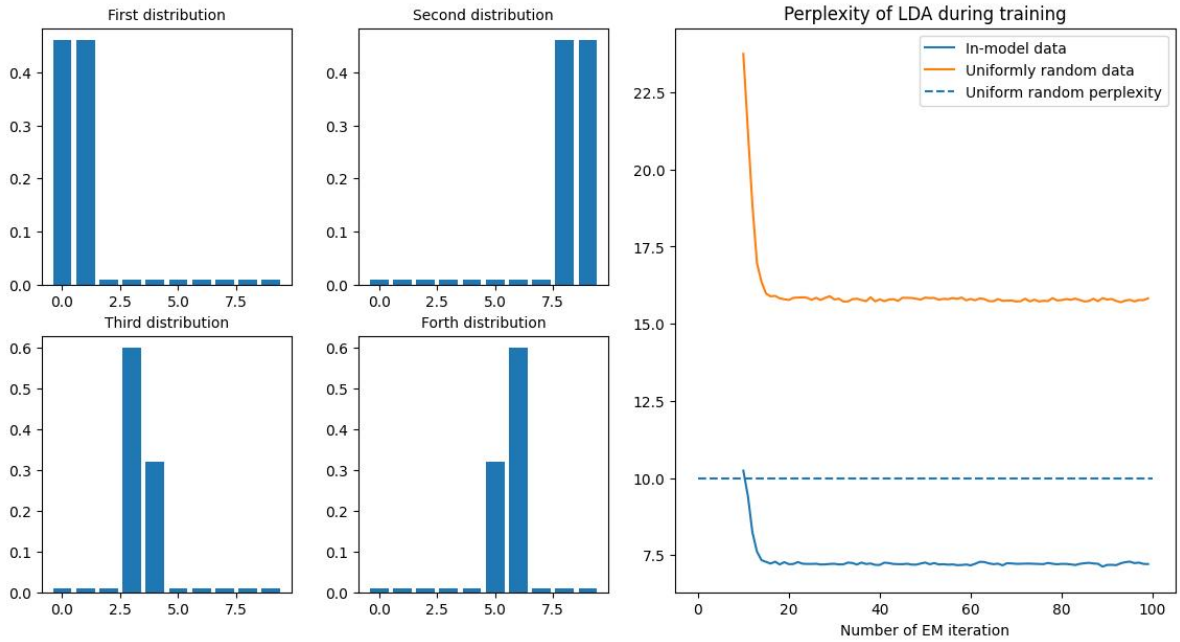

Figure 3: An experiment of measuring the perplexity as a function of number of variational EM iteration. The ground truth parameters of β are shown ($\alpha = [0.75, 0.75, 0.75, 0.75]$). Rights: The plot of perplexity of two datasets shows a much higher difference within themselves and between them and the $P = 10$ line.

on real data.

4.3 Document on simulated data

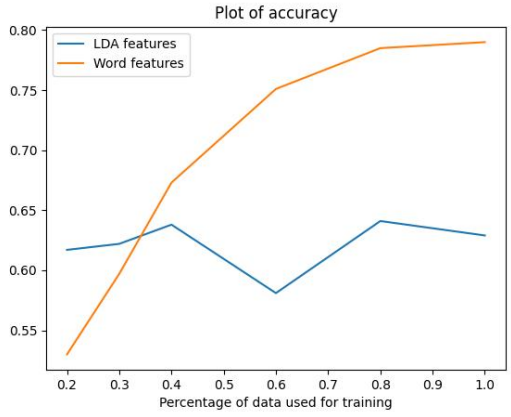

Figure 5: Plot of accuracy as a function of proportion of data used. The parameters are similar to figures 1 and 2. LDA based features manage to capture some information of the document, but lose to word features as more training data is used.

One method of classifying a document is to use the word frequencies as features, and then train supervided classification methods on this features-label dataset. The authors of [1] proposed another way: to first train an LDA on trained dataset without refering to the documents' label. Afterwards, given a document, the fitted LDA model would give a posterior distribution of the topics given the words in this document. We can approximate this distribution with a Dirichlet distribution using Variational methods (using the returned value γ from the algorithm 2), and use this as our new feature to perform supervised learning. This means that the feature size is reduces from the number of words in the dictionary (P) to the number of trained topic (K), which is much less in order of magnitude.

In our study, after generating data, we train SVMs to classify test data and measure its accuracy as a function of training data. For this experiment we also couldn't reproduce the authors result, as the accuracies for LDA features does not improve as more training data is used .

5 Conclusion

In this project, we study the Latent Dirichlet Allocation model as presented in [1]. LDA can be view as a dimensionality reduction technique, but is also a proper generative model capable of making sense of and simulating data). We use variation methods to derive inference and parameter estimation algorithm, and attempt to validate with numerical experiments on small-scale simulated data. We confirm mathematically and numerically the properties of these algorithms.

There are potential improvements that could be made to the project in the future. Better vectorized algorithmic implementations can help improving the speed of the model, allowing for testing on bigger simulated datasets or real datasets. Also, the approach of using Markov Chain Monte Carlo (MCMC) methods, such as the Gibbs sampler, to study LDA as an alternative for Variational methods is also a direction worth looking at.

LDA remains a very active field of research. Some directions include: spatial-aware LDA that can handle spatial information, LDA with a tree-based topic structure, LDA with sparse number of topics.

References

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993–1022, March 2003.

[2] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. Journal of the American society for information science, 41(6):391–407, 1990.

[3] Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. Genetics, 155(2):945–959, 06 2000.

[4] Gerard Salton and Michael McGill. Introduction to modern information retrieval. 1983.