# DD2434 Machine Learning, Advanced Course

Luca Marson, lmarson@kth.se
Alfredo Reichlin, alfrei@kth.se
Pei Yang Shi, pyshi@kth.se
Gianluigi Silvestri, giasil@kth.se
Group 2

January 21, 2018

**Abstract**

We chose the paper Latent Dirichlet Allocation by Blei et al (2003) [BNJ03]. We have implemented the algorithm suggested in the paper, but we found it to be too computationally expensive and not always reliable. We have also implemented an alternative approach - a collapsed Gibb's algorithm - for LDA. The results of both methods are presented in the following report.

# 1 Introduction

In this project, we decided to study and implement the contents of the article [BNJ03], that describes the probabilistic model Latent Dirichlet Allocation and the Variational Inference method used to estimate the posterior distribution. In addition to the methodology proposed in the paper we decided to use an alternative approach, based on the Gibbs sampling. In the following, we will describe the Latent Dirichlet Allocation and the implemented methods, providing also the obtained results and a final discussion.

# 2 Latent Dirichlet Allocation

Latent Dirichlet Allocation is an unsupervised generative probabilistic model for collections of discrete data such as text corpora. The model is able to categorize large collections of text into a finite set of topics. This could be useful for solving several problems such as classification, novelty detection, summarization, and similarity and relevance judgments.
The main idea behind LDA is that documents are modeled as mixtures of random variables over latent topics and each topic is itself a random distribution from which words are generated.
The model can be represented by the following graphical model:

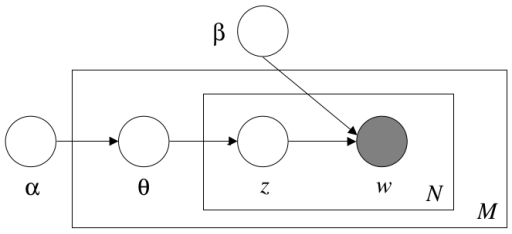


Figure 1: LDA graphical model

where the variables have the following meanings:

- $\theta$: per document topic proportion;
- $z_{d,n}$: per word topic proportion;
- $w_{d,n}$: observed word represented by a vector of length V, where V is the number of unique words in the corpus and which has a single component equal to one and all other components equal to zero;
- $\alpha$: proportions parameter;
- $\beta$: word probability treated as a fixed quantity.

The probability distributions over the variables are as follow:

- $\theta \sim Dir(\alpha)$
- $z_n \sim Multinomial(\theta)$
- $w_n \; is \; chosen \; from \; p(w_n|z_n, \beta)$
- $\beta_{i,j} = p(w^j = 1|z^i = 1)$

# 3 Methodology

## 3.1 Variational Inference

As described in Figure 1, the joint probability over the parameters for each document is:

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

In order to infer the LDA parameters, we need to compute the following posterior over the parameters:

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

Since this distribution is intractable, we can use variational inference to solve it by the defining the following alternative graphical model:

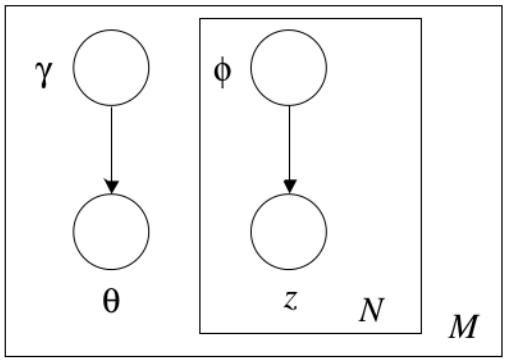


Figure 2: Graphical model of the variational distribution used to approximate the posterior

The posterior over the parameters is approximated by:

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

where $\gamma$ is a Dirichlet parameter and $\phi$ are multinomial parameters.

Our target is to find the approximate optimal parameters of the approximated posterior such that:

$$(\gamma^*, \phi^*) = argmin_{\gamma,\phi} KL(q(\theta, z|\gamma, \phi) \parallel p(\theta, z|w, \alpha, \beta))$$

As shown in [JGJS99], by using Jensen inequality the log likelihood of a document can be written as:

$$\log p(w|\alpha, \beta) = L(\gamma, \phi; \alpha, \beta) + KL(q(\theta, z|\gamma, \phi) \parallel p(\theta, z|w, \alpha, \beta)) \tag{1}$$

where $KL(q(\theta, z|\gamma, \phi) \parallel p(\theta, z|w, \alpha, \beta)$ is the Kullback-Leibler distance and $L(\gamma, \phi; \alpha, \beta)$ is a lower bound of the log likelihood.

Minimizing the KL distance is equivalent to maximizing the lower bound. As shown in [BNJ03], the optimal parameters are:

$$\phi_{ni} \propto \beta_{iw_n} \exp \Psi(\gamma_i) - \Psi(\sum_{j=1}^{K} \gamma_j)$$

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

Given a corpus of documents $D$ and the approximate optimal parameters $\gamma^*$, $\phi^*$, we need to compute the corpus level parameters $\alpha, \beta$, this can be achieved by maximizing the marginal log likelihood of the data:

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

Since this probability is intractable, by maximizing the function $L$ from equation (1), we can obtain the approximate optimal parameter $\alpha, \beta$:

$$\beta_{i,j} \propto \sum_{d=1}^{M} \sum_{n=1}^{N_d} \phi^*_{dni} w^j_{dn}$$

The approximate optimal $\alpha$ is the root of the derivative of the $L$ function with respect to $\alpha$ which is equal to:

$$M(\Psi(\sum_{j=1}^{K} \alpha_j) - \Psi(\alpha_i)) + \sum_{d=1}^{M} (\Psi(\gamma_{di}) - \Psi(\sum_{j=1}^{K} \gamma_{dj})$$

The root of this function can be efficiently computed using the Newton-Raphson algorithm.

The approximate optimal posterior parameters are thus evaluated through an alternating variational EM procedure, where in the E-step the optimal $\gamma, \phi$ are computed to maximize the lower bound $L$ from equation (1), while in the M-step the optimal $\alpha, \beta$ are computed.

## 3.2 Gibbs Sampling

After implementing the method in [BNJ03] with variational inference, our team decided to explore alternative ways to implement LDA. This was mainly due to the fact that the Variational Inference implementation was too slow to analyze big datasets. A notable paper we came across was [GS04], that described an alternative implementation in order to do inference on the posterior distribution of the LDA model with increased speed and obtaining meaningful results. The author described an approach using a classic MCMC algorithm - Gibbs [GRS95] algorithm that iteratively samples the assignment of topics to words in the various documents until convergence to the target distribution, and then uses new samples to do prediction on the several topics. The initialization of the procedure consists in randomly assigning each word of each document to one of the $K$ topics. The number of topics $K$ is chosen arbitrarily. This is done choosing the topic $k$ with an uniform distribution over the whole set of topics $K$. At each iteration of the Gibbs sampling we repeat the assignment of the topics sequentially for each topic. The probability $P(z_i \mid \boldsymbol{z_{-i}}, \boldsymbol{w})$ for each topic can be computed as:

$$P(z_i = j \mid \boldsymbol{z_{-i}}, \boldsymbol{w}) \propto \frac{n^{(w_i)}_{-i,j} + \beta}{n^{(\cdot)}_{i,j} + W\beta} \frac{n^{(d_i)}_{-i,j} + \alpha}{n^{(d_i)}_{i,\cdot} + T\alpha} \tag{2}$$

where $n^{(w_i)}_{-i,j}$ is the number of times that the word $w_i$ is assigned to the topic j without considering the assignment of $z_i$, $n^{(\cdot)}_{i,j}$ is the total number of words associated to topic j without considering the assignment of $z_i$, and the same principle applies for $n^{(d_i)}_{-i,j}$ and $n^{(d_i)}_{i,\cdot}$ where $d_i$ is a specific document. Alpha and Beta are hyper-parameters for the parameters of the prior and likelihood distributions. Gibbs sampling procedure are run a number of iterations, in our case 1000, by the end of which we can expect the Markov chain to converge at the target probability, and we can now use samples for inference.

## 3.3 Data Extraction

To run and test the implemented algorithms we have used the data set of the Associated Press composed of 2246 documents. Before executing the algorithms, we parsed the content of the documents and sanitized the data in order to obtain meaningful topic models. The operations executed are:

- Tokenizing: This step is required to convert the text into a series of strings corresponding to each word. The result of this operation is in fact a vector of words, one for each document, where the spaces and the punctuation have been removed;
- Stopping: Here we remove all the stop words that can't be classified in any particular topic. Examples of stop words in the English language are: "the","or","and".

- Stemming: The last step consists of remove all the words that have similar meaning and belong to the same root word. Examples can be the words "painted" "painter"and "painting", that can be reduced to the same word "paint". In this way, these words will be considered as equivalent entities, and they would have higher importance in the model.

These steps have been implemented in Python using the Natural language processing library"'NLTK", and the library for the stop words "stop_words". In [BNJ03], each word is represented by a vector of integers of length $V$, where $V$ is the number of words in the whole dictionary, and the $i-th$ word takes the value 1 in the position $v^i$ and 0 in all the other positions. For simplicity, we describe each word in the dictionary with an integer going from 0 to V-1.

# 4 Experiments and results

## 4.1 Results from Variational Inference

We have implemented the Variational Inference algorithm using Python. We have tried to test the code on the whole Associated Press dataset composed of 2246 documents, but the algorithm was too slow and the computational time taken by our computers would have been too high. Thus, to test the correctness of our code, we decided to used a reduced dataset with 300 documents, a number of topics $K = 20$. We believe that due to the small size of the data set, we didn't obtain satisfactory results. These results are shown in 4.1:

| topic 1 | topic 2 | topic 3 | topic 4 | topic 5 |
|---|---|---|---|---|
| police | american | company | party | three |
| said | one | million | soviet | york |
| soviet | new | three | said | last |
| people | bush | people | government | people |
| year | 000' | new | bush | new |
| will | will | percent | last | also |
| also | million | year | one | two |
| two | year | bush | day | said |
| new | people | said | year | percent |
| 000' | said | also | also | will |

Possible improvements could be trying further optimization of the code, implementing it in a more efficient language like C, or even trying to parallelize it using a GPU since the documents are mutually independent and therefore can be treated in parallel.

## 4.2 Results from Gibbs Sampling

The test here is done using the same database proposed in [BNJ03]. In particular is a reduced data set of the Associated Press composed of 2246 documents. With our implementation of the algorithm suggested by [GS04], we were able to achieve similar result as the variational inference approach. We tried the algorithm twice, and both times 1000 iterations were computed in a span of approximately 6 hrs. The hyper-parameters were $\alpha = 5$, $\beta = 0.1$ in one case, and in the other $\alpha = 0.1$, $\beta = 0.01$. The results of the first case are reported in 3, while the results for the second case are in 4. In particular, each column represent a different topic. We have selected 5 of the 100 topics obtained as result.

| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| budget | court | television | space | health |
| tax | case | news | shuttle | hospital |
| billion | judge | network | launch | medical |
| deficit | trial | time | earth | aids |
| income | charges | show | nasa | research |
| cute | attorney | week | mission | heart |
| spending | district | tv | columbia | disease |
| services | federal | abc | test | dr |
| taxes | jury | cbs | make | study |
| less | filed | nbc | lost | treatment |
| year | charged | night | set | used |
| congress | hearing | broadcast | venus | blood |
| social | former | series | launched | patients |
| security | ruling | 14 | rocket | center |
| interest | alleged | prime | ground | doctors |
| reduction | convicted | shows | feet | researchers |
| cuts | lawyers | season | spacecraft | virus |
| 1991 | accused | roberts | magellan | body |
| programs | guilty | rating | explosion | cancer |
| poor | testimony | sunday | radar | use |

Figure 3: $\alpha = 5$, $\beta = 0.1$

| topic 1 | topic 3 | topic 4 | topic 6 | topic 7 |
|---|---|---|---|---|
| tax | budget | iran | species | san |
| income | billion | hostages | animals | california |
| trust | bush | iranian | park | los |
| taxes | deficit | release | fish | angeles |
| federal | congress | held | wildlife | francisco |
| irs | president | hijackers | birds | heat |
| keating | administration | islamic | endangered | record |
| returns | spending | lebanon | animal | summer |
| property | year | hostage | monet | thursday |
| estate | cut | thursday | wild | calif |
| payments | new | american | yosemite | county |
| lincoln | programs | red | state | diego |
| deconcini | government | arms | oregon | santa |
| corporations | cuts | anderson | summer | cities |
| april | security | geneva | eggs | beach |
| paid | tax | tehran | trees | friday |
| regulators | reduction | gunmen | environmental | high |
| taxpayers | program | de | turtles | la |
| senators | taxes | 1985 | habitat | air |
| five | fiscal | turkish | service | miami |

Figure 4: $\alpha = 0.1$, $\beta = 0.01$

To show the convergence of the Gibbs sampling,in the following pictures is shown the improvement of the logarithm of the posterior distribution over the iterations 5.

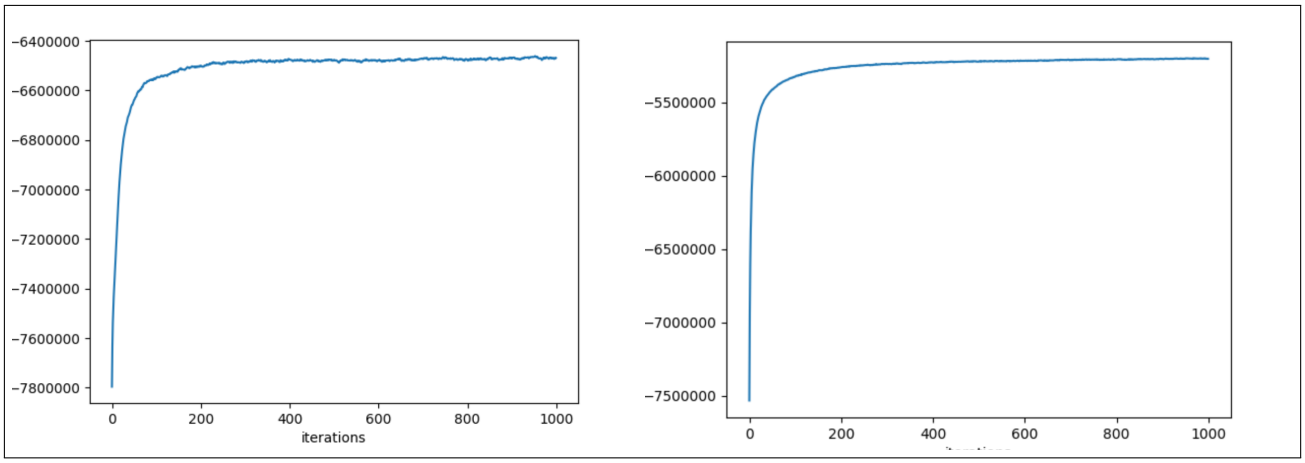


Figure 5: left: $\alpha = 5$, $\beta = 0.1$, right: $\alpha = 0.1$, $\beta = 0.01$

# 5 Discussion and conclusion

This was a comprehensive project implementing and comparing the two approaches to achieve Latent Dirichlet Allocation.
LDA can be an effective model for text corpora analysis, but we found that parameters estimation through variational inference can be computationally expensive.

Our implementation of collapsed Gibb's algorithm [GS04] was considerably simpler than the Variational inference approach. Because of the nature that MCMC algorithms which do not assume the underlying distribution, with enough simulation it can very closely model the target distribution. We have achieved very similar results as Variational Inference approach, and much faster processing time comparing to our . We recommend using the collapsed Gibb's sampling for very complex features or large data set.

Latent Dirichlet Allocation is an efficient and effective system of discovering topics with potential keywords. We are very eager to see

# References

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[GRS95] W.R. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC Interdisciplinary Statistics. Taylor & Francis, 1995.

[GS04] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.

[JGJS99] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Mach. Learn.*, 37(2):183–233, November 1999.