Topic modeling: Latent Dirichlet Allocation

Sharat Chikkerur

Textual representation

- Goals
  - Compactness
  - Generalization
  - Semantic interpretation
- Methods
  - Bag of words, TF IDF (Term frequency - Inverse document frequency)
  - LSI (Latent Semantic Indexing)
  - Probabilistic Topic modeling
    - Topic mixture Models
    - Latent Semantic Indexing
    - Latent Dirichlet Allocation

- Vector embedding
  * word2vec
  * GloVe

Bag of words

Procedure to obtain bag-of-words representation:

- A dictionary of tokens is generated using text from the entire corpus. This defines an ordered collection of words $w_1, w_2 \dots w_V$.
- Each document is represented using a vector $\mathbf{n} = [n_{w_i}], i \in [1 \dots N]$ consisting of frequencies of each word in the dictionary
- This generates a sparse representation of each text – still high dimensional ($|V|$).
- The dictionary is usually pre-processed to remove stop-words (for, the, is etc.) and also words with very-high (non informative) and very low frequencies (does not generalize).
- Example: Consider dictionary "A", "B", "C", "D" and the document "A A B C". The BoW represetation will be ["A": 2, "B": 1, "C": 1, "D": 0]
- All positional information about words is lost: BagOfWords("I have a bag of word") = BagOfWords("words of bag have I")

TF-IDF (Term frequency - Inverse document frequency)

- Bag of words accounts only for term frequency (relative frequency of a word within a document), but does not consider its relative frequency within the corpus.

- The idea behind TF-IDF is to weight each word by its relative rarity (inverse document frequency).

- Given a vocabulary of words $w_1, w_2 \ldots w_V$, we inverse document frequency table

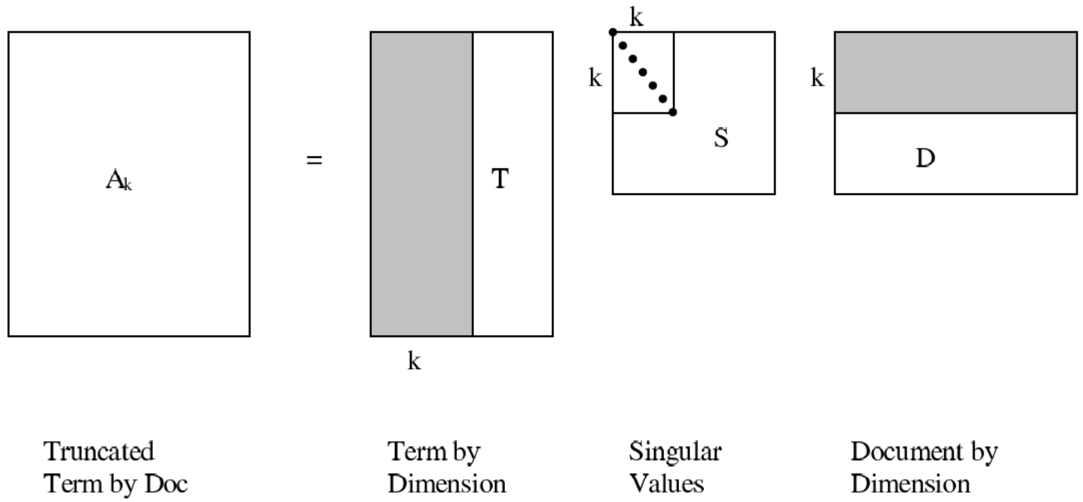
$$D_{w_i} = \frac{\text{Total number of documents}}{\text{Number of documents with the given word}}$$

- For each document, we compute a local frequency table n_{w_i} as before.

- TF-IDF $\sim \frac{n_{w_i}}{log(D_{w_i})}$

Latent Semantic Indexing

- Bag of words and TF-IDF representation is sparse but high dimensional.
- If we treat TF-IDF representation of a corpus as a matrix, we can use SVD to get a lower dimensional representation.



- Each document can now be represented using it's scale vector.

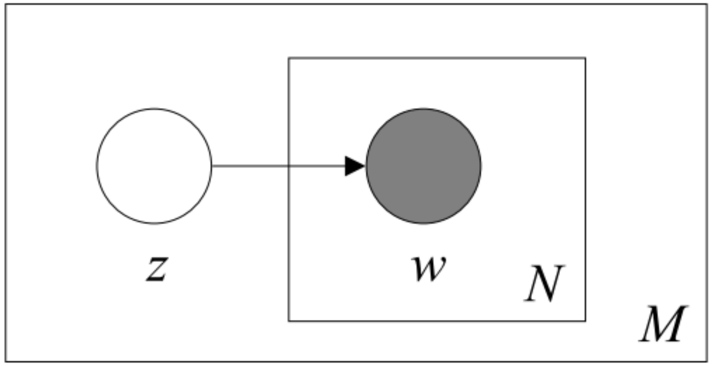
http://www.dimacs.rutgers.edu/~billp/pubs/IPM.pdf

Latent Semantic Indexing (cont.)

- Each basis vector can be thought of as a 'topic' with some semantic meaning.
- Does not enforce exclusivity of words within each topic.
- Generates a dense lower dimensional representation of each document

Probabilistic topic mixture model

Generative model

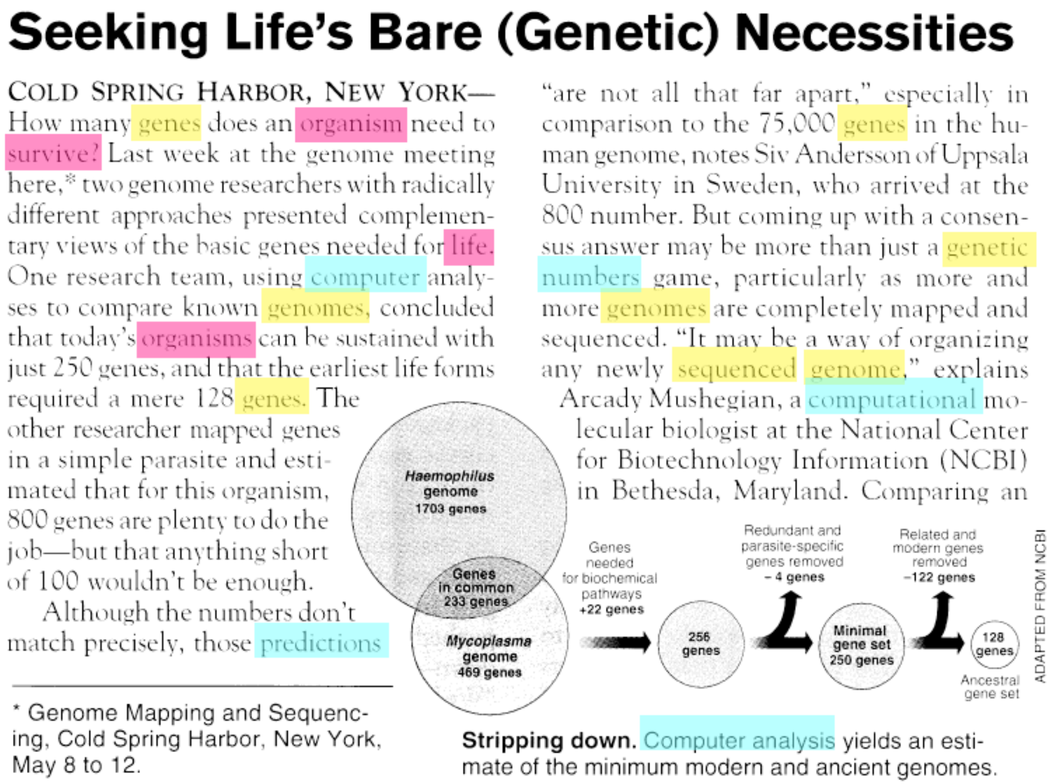


- For each document d pick a topic $z_d \sim Multinomial(\beta)$
- For each word w_i in the document pick a word $w_i \sim Multinomial(\beta_{z_d})$
- The vector $[p(z_1) \dots p(z_K)]$ provides a compact representation for each document.

$$p(w) = \sum_z p(z) \prod_n p(w_n|z)$$

This is similar in spirit to GMM for numeric data.

Topic modeling



Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

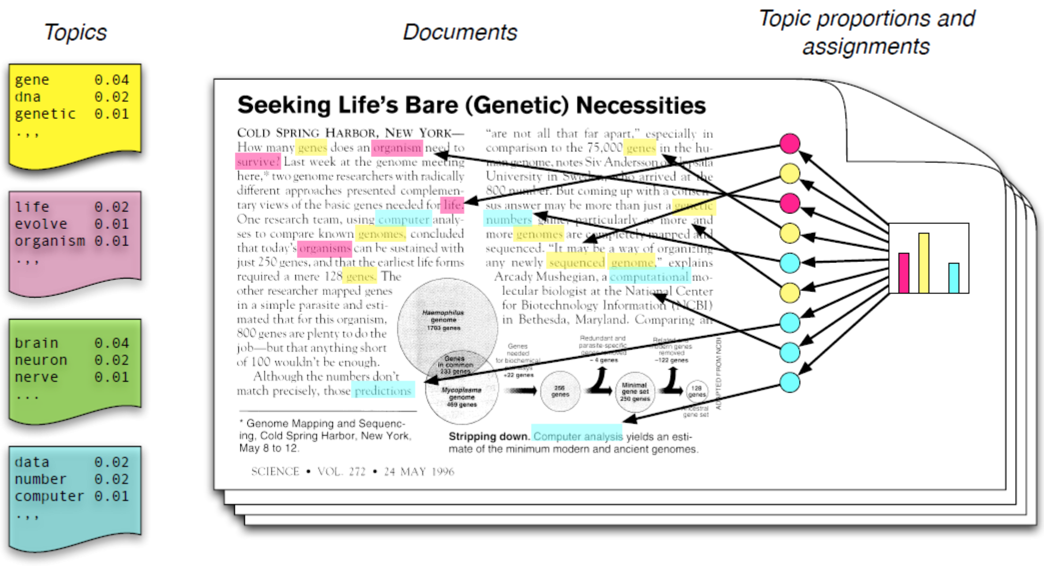
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

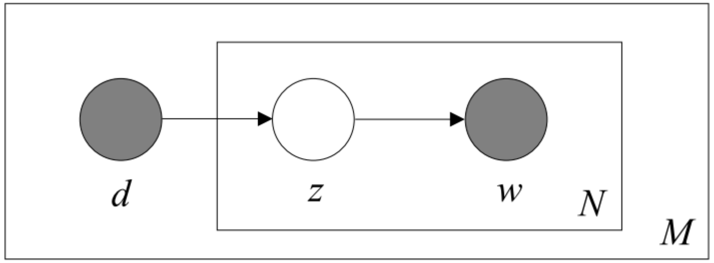
Each document is a mixture of topics

Topic modeling (cont.)



- Each topic is a distribution over words
- Each word is drawn from one of the topics

Probablistic latent Semantic Indexing

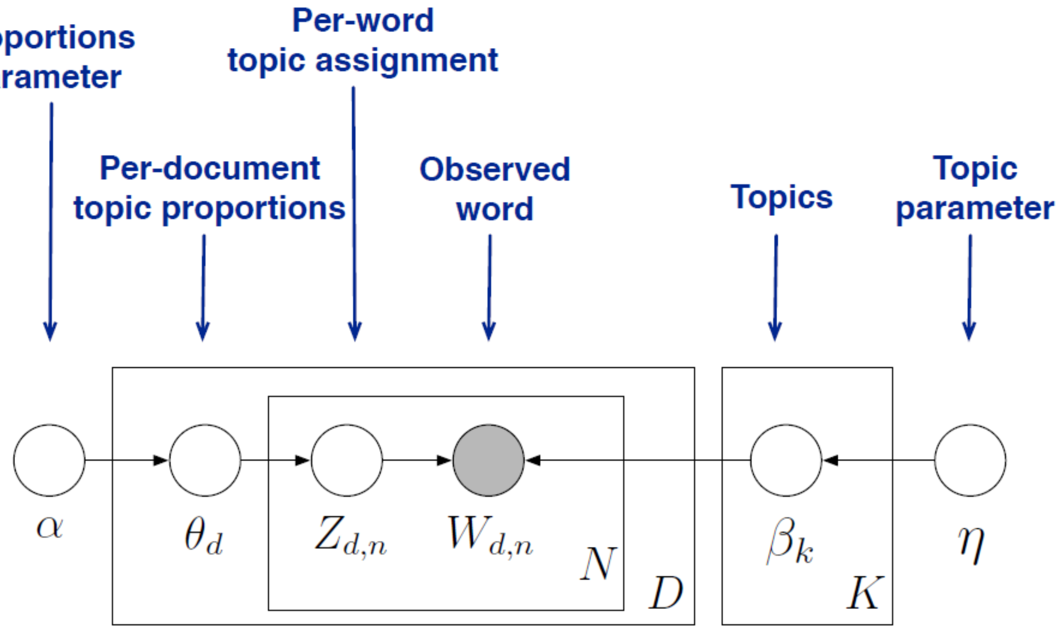


Generative model

- Pick a document with probability $p(d)$.
- For each word w_i in the document pick a topic with probability $p(z|d)$, Sample the word $w_i \sim Multinomial(\beta_{z_d})$
- The vector $[p(z_1|d) \dots p(z_K|d)]$ provides a compact representation for each document.
- Different from mixture model: topic is sampled for each **word** instead of for each document.

$$p(d, w_n) = p(d) \sum_z p(z|d) p(w_n|z)$$

Latent Dirichlet Allocation



$$p(\beta, \theta, \mathbf{z}, \mathbf{w}) = \left(\prod_{i=1}^{K} p(\beta_i \mid \eta)\right) \left(\prod_{d=1}^{D} p(\theta_d \mid \alpha) \prod_{n=1}^{N} p(z_{d,n} \mid \theta_d) p(w_{d,n} \mid \beta_{1:K}, z_{d,n})\right)$$

Latent Dirichlet Allocation

Generative model

- Pick a topic distribution $\theta \sim Dir(\alpha)$.
- For each word w_i in the document
  - choose a topic $z_i \sim$ Multinomal(θ)
  - choose a word from $p(w_i|z_i, \beta)$

Note:

- $p(w_n|\theta, \beta)$ is a random variable since it depends on θ
- $p(w|\alpha, \beta) = \int \{p(\theta|\alpha) \prod_i p(w_i|\theta, \beta)\} \mathrm{d}\theta$

Side note: Dirichlet distribution

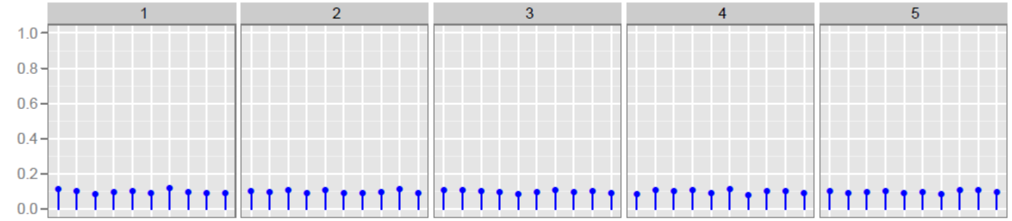
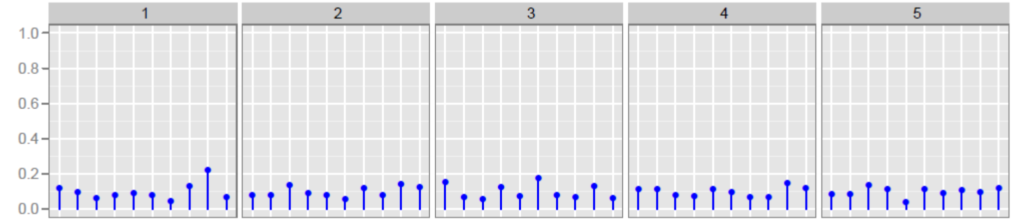
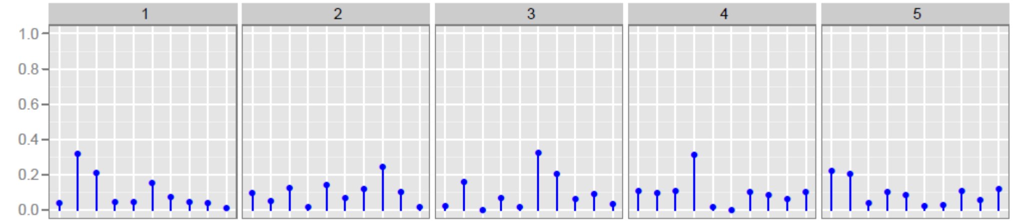
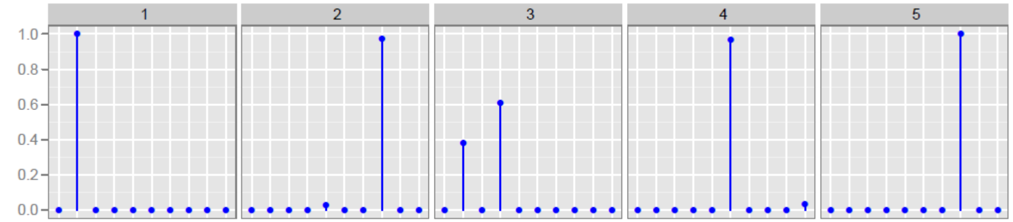
- Defines a distribution over the simplex (multinomial)

$$p(\theta|\vec{\alpha}) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

- Each sample from this distribution is a multinomial distribution

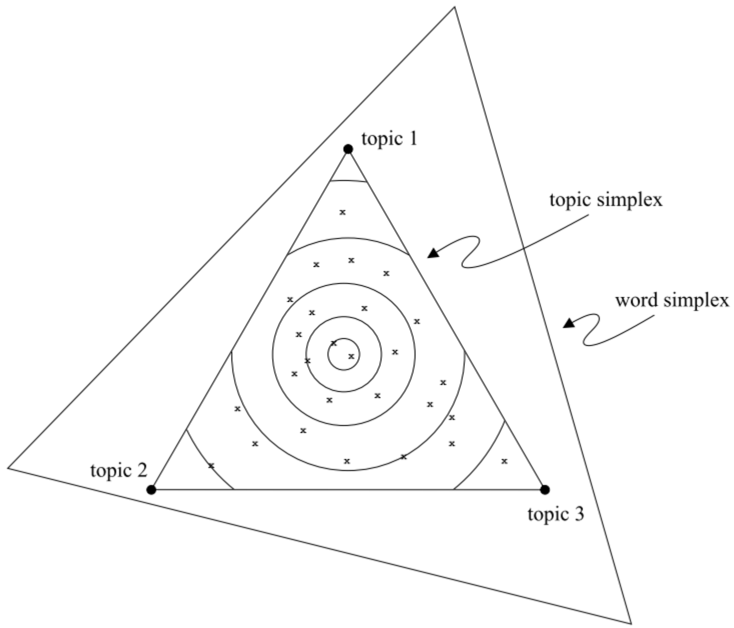
- It is also the conjugate to the multinomial

Side note: Effect of prior

$\alpha = 100$	
$\alpha = 10$	
$\alpha = 1$	
$\alpha = 0.1$	

Geometric perspective

Comparison with other methods:



Inference and learning

- Key inference problem in LDA is computing the distribution of hidden variables given a document.

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

- This is intractable in general.

- We have to resort to approximation methods.

Approximate Inference

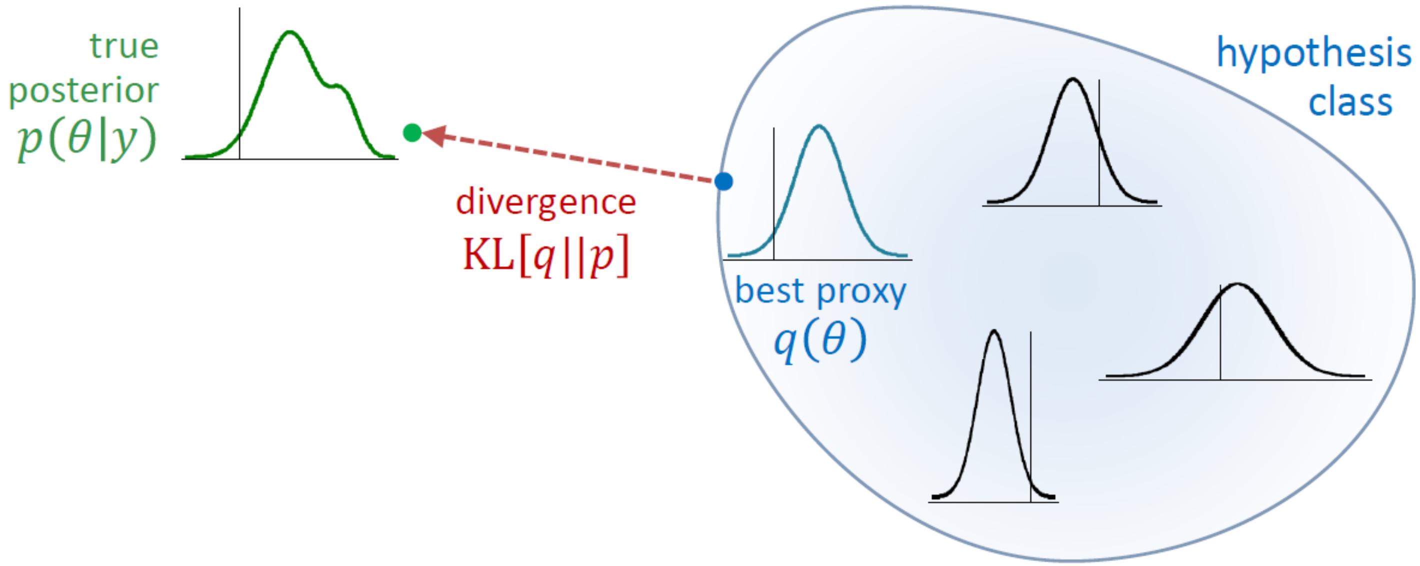
- Mean field variational methods
- Expectation propagation
- MCMC - Collapsed Gibbs Sampling
- MCMC - Distributed sampling
- Factorization based inference
- **Online variational inference**

Variational Bayes

- Variational Bayes is a generalization of Laplace approximation
- Instead of evaluating exact distribution, we approximate it using a family of distribution (e.g. mixture)
- The substitute family of distributions are easier to compute
- We find the distribution that is 'closest' (in terms of KL divergence) to the exact distribution.

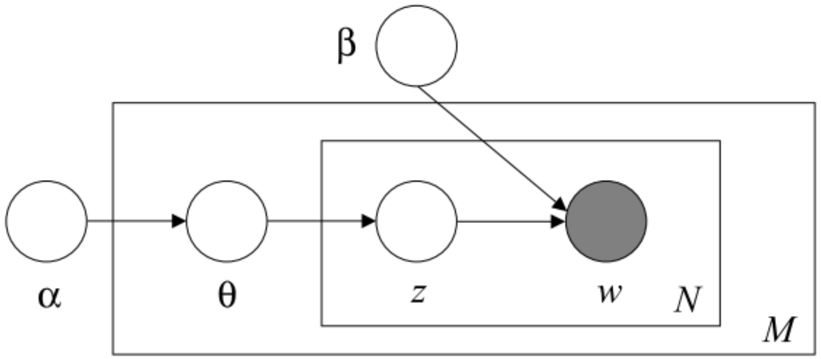
Variational Bayes (cont)

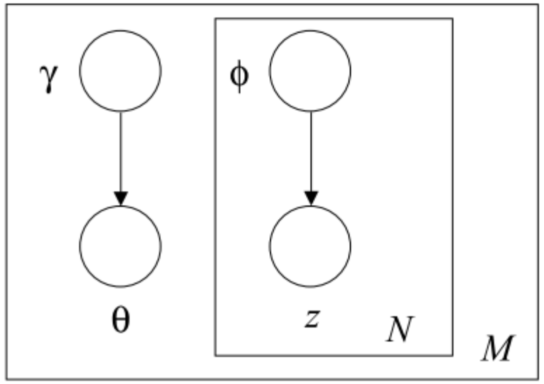
Example:



http://people.inf.ethz.ch/bkay/talks/Brodersen_2013_03_22.pdf

LDA: Variational Bayes





The exact posterior,

$$p(\theta, z|w, \alpha, \beta)$$

is replaced by a parametric distribution:

$$q(\theta, z|\gamma, \phi) = q(\theta|\gamma) \prod_n q(z_n|\phi_n)$$

Optimization consists of finding

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

Batch variational Bayes Optimization

Algorithm 1 Batch variational Bayes for LDA

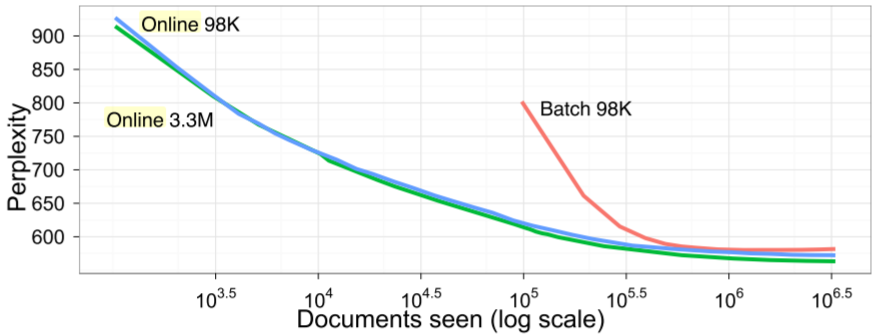
Initialize $\boldsymbol{\lambda}$ randomly.
while relative improvement in $\mathcal{L}(\boldsymbol{w}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\lambda}) > 0.00001$ **do**
  E step:
  for $d = 1$ to D **do**
    Initialize $\gamma_{dk} = 1$. (The constant 1 is arbitrary.)
    repeat
      Set $\phi_{dwk} \propto \exp\{\mathbb{E}_q[\log \theta_{dk}] + \mathbb{E}_q[\log \beta_{kw}]\}$
      Set $\gamma_{dk} = \alpha + \sum_w \phi_{dwk} n_{dw}$
    until $\frac{1}{K} \sum_k |\text{change in} \gamma_{dk}| < 0.00001$
  end for
  M step:
  Set $\lambda_{kw} = \eta + \sum_d n_{dw} \phi_{dwk}$
end while

Online variational Bayes Optimization

Algorithm 2 Online variational Bayes for LDA

Define $\rho_t \triangleq (\tau_0 + t)^{-\kappa}$
Initialize $\boldsymbol{\lambda}$ randomly.
for $t = 0$ to ∞ **do**
    E step:
    Initialize $\gamma_{tk} = 1$. (The constant 1 is arbitrary.)
    repeat
        Set $\phi_{twk} \propto \exp\{\mathbb{E}_q[\log \theta_{tk}] + \mathbb{E}_q[\log \beta_{kw}]\}$
        Set $\gamma_{tk} = \alpha + \sum_w \phi_{twk} n_{tw}$
    until $\frac{1}{K} \sum_k |\text{change in} \gamma_{tk}| < 0.00001$
    M step:
    Compute $\tilde{\lambda}_{kw} = \eta + D n_{tw} \phi_{twk}$
    Set $\boldsymbol{\lambda} = (1 - \rho_t)\boldsymbol{\lambda} + \rho_t \tilde{\boldsymbol{\lambda}}$.
end for

Evaluation



Documents analyzed	**2048**	**4096**	**8192**	**12288**	**16384**	**32768**	**49152**	**65536**
Top eight words	systems	systems	service	service	service	business	business	business
	road	health	systems	systems	companies	service	service	industry
	made	communication	health	companies	systems	companies	companies	service
	service	service	companies	business	business	industry	industry	companies
	announced	billion	market	company	company	company	services	services
	national	language	communication	billion	industry	management	company	company
	west	care	company	health	market	systems	management	management
	language	road	billion	industry	billion	services	public	public

Perplexity: $\exp\{-\frac{\Sigma_d \log p(\mathbf{w}_d)}{\Sigma_d N_d}\}$

VW LDA

- **Input format**

```
| no label required
| does not support namespace
| can:0 contain:2 counts:10
```

- **Compatibility**
  - `"--audit"` does not work
  - `"--invert_hash"` does not work

VW LDA, Options

- `--lda <n>` : number of topics
- `--lda_D <n>`: approximate number of documents
- `--lda_alpha <n>`: Dirichlet prior on topics
- `--lda_rho <n>`: Dirichlet prior on word
- `--minibatch <n>`: Size of the minibatch

Demo

References

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2012. ISSN 15324435. doi: 10.1162/jmlr.2003.3.4-5.993. URL http://www.cs.princeton.edu/ blei/lda-c/\npapers2://publication/doi/10.1162/jml

Matthew D Hoffman, David M Blei, and Francis Bach. Online Learning for Latent Dirichlet Allocation. *Advances in Neural Information Processing Systems*, 23:1–9, 2010. ISSN 08912017. doi: 10.1145/1835804.1835928. URL http://books.nips.cc/papers/files/nips23/NIPS2010_1291.pdf.