

Topic modelling.

Victor Kitov

v.v.kitov@yandex.ru

Common dimensionality reduction methods

- LSA
- non-negative matrix factorization
- pLSA
- LDA
- more advanced topic models

LSA

- LSA (latent semantic analysis) - the process of applying SVD for dimensionality reduction
 - mainly used in text domain (objects=documents)
 - also called LSI (latent semantic indexing)
- SVD decomposition: $X = U\Sigma V^T$, $U^T U = I$, $V^T V = I$, $\Sigma = \text{diag} \{ \sigma_1^2, \dots, \sigma_R^2 \}$, $U, V \in \mathbb{R}^{N \times R}$, $\Sigma \in \mathbb{R}^{R \times R}$, $R = \text{rg } X$
- Truncated SVD of order K :
 - $\hat{X}_K = U_K \Sigma_K V_K^T$, U_K, V_K -first K columns of U, V ; Σ_K -first K columns&rows of Σ
 - $U_K, V_K \in \mathbb{R}^{N \times K}$, $\Sigma_K \in \mathbb{R}^{K \times K}$, $K \leq R$, usually $K \in [200, 500]$.

LSA

- Property of truncated SVD:
 - $\hat{X}_K = \arg \min_{B: \text{rg } B \leq K} \|X - B\|_{\text{Frobenius}}^2$
- Low order representations for new objects $x \in \mathbb{R}^{1 \times D}$
 - $U = XV\Sigma^{-1} \Rightarrow u = xV\Sigma^{-1}$
 - $U_K = XV_K\Sigma_K^{-1} \Rightarrow u_K = xV_K\Sigma_K^{-1}$

pLSA¹

- pLSA = probabilistic latent semantic analysis
- It is a probabilistic generative model for words in documents

¹Thomas Hofmann, Probabilistic Latent Semantic Indexing, SIGIR-99, 1999.

pLSA definition

- Documents collection may be represented as a sequence $\langle d, w_{d,c} \rangle_{d=1, D}^{c=1, n_d}$ of <document, word> pairs, where
 - d - document number
 - c - word-position number inside document
 - n_d - length of document d
- We will have $n = \sum_{d=1}^D n_d$ such pairs.

pLSA generation

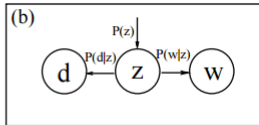
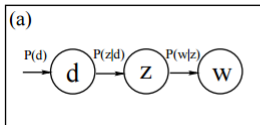
- For each word position:
 - document is sampled $d \sim p(d)$
 - unobserved topic is sampled $z \sim p(z|d)$
 - word is sampled $w \sim p(w|z)$
- Comments:
 - Each document defines some distribution on topics $z \sim p(z|d)$
 - Each topic defines a distribution on words $w \sim p(w|z)$
 - only topic affects word distribution (not $\langle \text{topic}, \text{document} \rangle$)

Equivalent «symmetric» representation of pLSA

$$p(d, w) = p(d)p(w|d) = p(d) \sum_z p(z|d)p(w|z) \quad (1)$$

$$= \sum_z p(d, z)p(w|z) = \sum_z p(z)p(d|z)p(w|z) \quad (2)$$

(a) asymmetric and (b) symmetric pLSA representation



Connection of pLSA to LSA

- In matrix form $X = U\Sigma V^T$, where
 - $X \in \mathbb{R}^{D \times W}$, $U \in \mathbb{R}^{D \times K}$, $\Sigma \in \mathbb{R}^{K \times K}$, $V \in W \times K$
 - U, V - are stochastic, not orthogonal matrices
 - U, Σ, V are estimated with maximum likelihood, not Frobenius norm minimization.
- pLSA - more interpretable
 - document-topics distribution
 - topic-word distribution
 - We can truncate this representation by taking only topics with $p(z) \geq \text{threshold}$.
 - allows finding semantically close documents **and words**
 - segmentation into topics of running text

Dimensionality reduction with pLSA

- Define $x_{dw} := p(w|d)$, $a_{dz} := p(z|d)$, $b_{zw} := p(w|z)$
- $X = \{x_{dw}\} \in \mathbb{R}^{D \times W}$, $A = \{a_{dz}\} \in \mathbb{R}^{D \times K}$, $B = \{b_{zw}\} \in \mathbb{R}^{K \times W}$
- $p(w|d) = \sum_z p(z|d)p(w|z)$
- In matrix form $X = AB$
- $a_{d,:} \in \mathbb{R}^K$ -low dimensional representation of document d
- $b_{:,w} \in \mathbb{R}^K$ -low dimensional representation of word w
- Allows to find similar/dissimilar documents and words.

Segmentation into topics of running text

Label words with

$$\arg \max_z p(z|d, w) = \arg \max_z \frac{p(z, d, w)}{p(d, w)} = \arg \max_z p(z)p(d|z)p(w|z)$$

Topics

gene 0.04
dna 0.02
genetic 0.01
...

life 0.02
evolve 0.01
organism 0.01
...

brain 0.04
neuron 0.02
nerve 0.01
...

data 0.02
number 0.02
computer 0.01
...

Documents

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**

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

SCIENCE • VOL. 272 • 24 MAY 1996

"are not all that far apart," especially in comparison to the 75,000 **genes** in the human genome, notes Siv Andersson, a geneticist at the University in Sweden who arrived at the 800 number. But coming up with a **compact** answer may be more than just a **genetic** numbers game, particularly as more and more **genomes** are completely sequenced and analyzed. "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



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

Topic proportions and assignments



Counters definitions

- n - total number of word positions
- n_d - number of word positions in document d
- n_{dwz} -number $\langle d, w, z \rangle$ triples
- $n_{dw} = \sum_z n_{dwz}$, $n_{dz} = \sum_w n_{dwz}$, $n_{wz} = \sum_d n_{dwz}$
- $n_d = \sum_w n_{dw} = \sum_z n_{dz}$, $n_w = \sum_d n_{dw} = \sum_z n_{wz}$,
 $n_z = \sum_d n_{dz} = \sum_w n_{wz}$

General probabilistic model with latent variables

Suppose objects have observed features x and unobserved (latent) features z .

- $[x, z] \sim p(x, z, \theta)$, $x \sim p(x, \theta)$
- denote $X = [x_1, x_2, \dots, x_N]$, $Z = [z_1, z_2, \dots, z_N]$.

To find $\hat{\theta}$ we need to solve

$$L(\theta) = \ln p(X|\theta) = \ln \sum_Z p(X, Z|\theta) \rightarrow \max_{\theta}$$

- This is intractable for unknown Z .
- We need to fallback to iterative optimization, such as SGD.
- Alternatively, we may use EM algorithm, which “averages” over different fixed variants of Z .

EM algorithm

INPUT:training set $X = [x_1, \dots, x_N]$ some initialization **for** $\hat{\theta}$

some predefined convergence criteria

ALGORITHM:**repeat until** convergence:

E-step: set distribution over latent variables:

$$q(Z) = p(Z|X, \theta)$$

M-step: improve estimate of θ

$$\hat{\theta} = \arg \max_{\theta} \{ \sum_Z q(Z) \ln p(X, Z|\theta) \}$$

OUTPUT:ML estimates $\hat{\theta}$ **for** training set.

EM algorithm for pLSA²

- Define n_{dw} -the count of word w in document d , c -cell number c in document, filled with some word w . Document d has n_d words (=cells filled with these words).
- Parameters
 $\theta = \{p(z), p(d|z), p(w|z); z = \overline{1, K}, w = \overline{1, W}, d = \overline{1, D}\}$
- Likelihood:

$$\begin{aligned}
 P(\{d, w_{d,c}\}_{c=\overline{1, n_d}}^{d=\overline{1, D}}) &= \prod_{d=1}^D \prod_{c=1}^{n_d} p(d, w_{d,c}) \\
 &= \prod_{d=1}^D \prod_{c=1}^{n_d} \sum_z p(z) p(w_{d,c}|z) p(d|z)
 \end{aligned}$$

²Derive pLSA estimation with EM when each document may belong only to single topic.

EM algorithm for pLSA

- Log-likelihood (direct maximization intractable):

$$\ln P(\{d, w_{d,c}\}_{c=1, n_d}^{d=1, D}) = \sum_{d=1}^D \sum_{c=1}^{n_d} \ln \left(\sum_z p(z) p(w_{d,c}|z) p(d|z) \right)$$

- For known $z_{d,c}$ for all word positions:

$$\ln P(\{d, w_c\}_{c=1, n_d}^{d=1, D}) = \sum_{d=1}^D \sum_{c=1}^{n_d} \ln [p(z_{d,c}) p(w|z_{d,c}) p(d|z_{d,c})]$$

EM algorithm for pLSA

- E-step:

$$\begin{aligned} p(z|d, w) &= \frac{p(z, d, w)}{p(d, w)} = \frac{p(z)p(d|z)p(w|z, \mathcal{d})}{\sum_{z'} p(z')p(d|z')p(w|z', \mathcal{d})} \\ &= \frac{p(z)p(d|z)p(w|z)}{\sum_{z'} p(z')p(d|z')p(w|z')} \end{aligned}$$

- M-step (reestimation of θ using $p(z|d, w)$):

$$\begin{aligned} &\mathbb{E}_z \ln P(\{d, w_c, z_c\}_{c=1, n_d}^{d=1, D}) \\ &= \sum_{d=1}^D \sum_{c=1}^{n_d} \sum_z p(z|d, w_{d,c}) \ln [p(z)p(w_{d,c}|z)p(d|z)] \\ &= \sum_{d=1}^D \sum_{w=1}^W n_{dw} \sum_z p(z|d, w) \ln [p(z)p(w|z)p(d|z)] \end{aligned}$$

EM algorithm for pLSA

Constraints:

$$\sum_z p(z) = \sum_w p(w|z) = \sum_d p(d|z) = 1$$

Lagrangian:

$$L = \sum_{d=1}^D \sum_{w=1}^W n_{dw} \sum_z p(z|d, w) [\ln p(z) + \ln p(w|z) + \ln p(d|z)]$$

$$+ \alpha \sum_z (1 - p(z)) + \beta_z \sum_w (1 - p(w|z)) + \gamma_z \sum_d (1 - p(d|z))$$

$$\frac{\partial L}{\partial p(z)} = \sum_d \sum_w n_{dw} p(z|d, w) \frac{1}{p(z|w)} - \alpha = 0$$

$$p(z) \propto \sum_d \sum_w n_{dw} p(z|d, w) = n_z$$

$$p(z) = n_z/n$$

EM algorithm for pLSA

$$\frac{\partial L}{\partial p(w|z)} = \sum_d n_{dw} p(z|d, w) \frac{1}{p(w|z)} - \beta_z = 0$$

$$p(w|z) \propto \sum_d n_{dw} p(z|d, w) = n_{wz}$$

$$p(w|z) = n_{wz}/n_z$$

$$\frac{\partial L}{\partial p(d|z)} = \sum_w n_{dw} p(z|d, w) \frac{1}{p(d|z)} - \gamma_z = 0$$

$$p(d|z) \propto \sum_w n_{dw} p(z|d, w) = n_{dz}$$

$$p(d|z) = n_{dz}/n_z$$

EM algorithm

Initialize $p(z)$, $p(d|z)$, $p(w|z)$.

Iterate until convergence:

- E-step:

$$p(z|d, w) = \frac{p(z)p(d|z)p(w|z)}{\sum_{z'} p(z')p(d|z')p(w|z')}$$

- M-step:

$$n_z = \sum_d \sum_w n_{dw} p(z|d, w)$$

$$n_{wz} = \sum_d n_{dw} p(z|d, w)$$

$$n_{dz} = \sum_w n_{dw} p(z|d, w)$$

$$p(z) = \frac{n_z}{n} \quad p(w|z) = \frac{n_{wz}}{n_z} \quad p(d|z) = \frac{n_{dz}}{n_z}$$

Comments

- $p(z|d)$ - more reasonable and useful statistic.

$$\begin{aligned}
 p(z|d) &= \frac{p(z, d)}{p(d)} = \frac{p(z)p(d|z)}{\sum_{z'} p(z')p(z'|d)} = \frac{\frac{n_z}{n} \frac{n_{dz}}{n_z}}{\sum_{z'} \frac{n_{z'}}{n} \frac{n_{dz'}}{n_{z'}}} \\
 &= \frac{n_{dz}}{\sum_{z'} n_{dz'}} = \frac{n_{dz}}{n_d}
 \end{aligned}$$

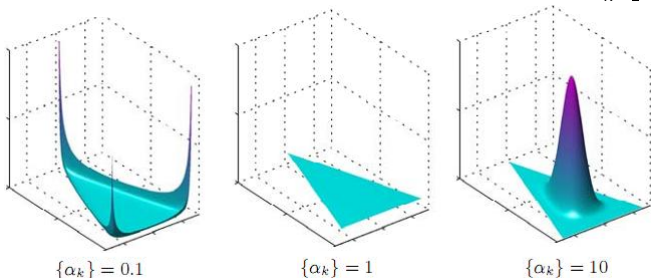
- Result of EM depends on starting conditions!
 - better init with some reasonable topics, based on human document categorization.
 - may init using results of document clustering into K -clusters
 - more natural to init $p(z|d)$ than $p(d|z)$

LDA method³

- Bayesian extension of pLSA
- Distributions $p(z|d)$ and $p(w|z)$ are «inner random parameters» with prior distributions:

$$p(z|d) \sim \text{Dir}(\alpha), \quad p(w|z) \sim \text{Dir}(\beta)$$

Probability density function of Dirichlet(α), $\alpha = \{\alpha_k\}_{k=1}^K$



³Derive LDA estimation with EM for $\alpha \geq 1$, $\beta \geq 1$ elementwise.

LDA variables

Parameters:

- α -Dirichlet prior on topics distributions $p(z|d)$
- β -Dirichlet prior on words distributions $p(w|z)$

Estimated values:

- $\varphi_z = p(w|z)$, $w = \overline{1, W}$, $z = \overline{1, Z}$
- $\theta_d = p(z|d)$, $z = \overline{1, Z}$, $d = \overline{1, D}$

Latent variables:

- topics at each word-position:

$$z_i^d, \quad d = \overline{1, D}, i = \overline{1, n_d}$$

Observed variables:

- words at each word-position:

$$w_i^d, \quad d = \overline{1, D}, i = \overline{1, n_d}$$

LDA-data generation process

- 1 generate $\theta_d \sim \text{Dir}(\alpha)$, $d = \overline{1, D}$
- 2 generate $\varphi_z \sim \text{Dir}(\beta)$, $z = \overline{1, Z}$
- 3 for each document d and each word-position $n = \overline{1, n_d}$:
 - 1 generate topic $z_n^d \sim \text{Multinomial}(\theta_d)$
 - 2 generate word $w_n^d \sim \text{Multinomial}(\varphi_{z_n^d})$

Extensions of topic models

- Automatically select number of topics (e.g. HDP)
 - still need to specify «willingness to make new topic»
- hierarchical set of topics
 - greedy layerwise optimization
 - joint optimization for whole hierarchy

Extensions of topic models

- Document representation may contain not only sequence of words but also sequence of other entities:
 - possible entities: title text, authors, keywords, links, users who read them, etc.
- If document has other properties, they can be also generated in topic model
 - length, time, source, etc.
- Topic modeling can be applied to any objects, represented as sequences of entities
 - we considered only objects=documents, entities=words on word-positions.
- Other possible application domains:
 - DNA sequences of genes
 - video records with particular events

Applications⁴

- Topic models can be applied for:
 - dimensionality reduction (feature extraction)
 - document clustering
 - document summarization
 - topics segmentation inside documents
 - word clustering

⁴See K.V.Voronsov's course on topic modelling for more.