Topic Model (1): pLSI

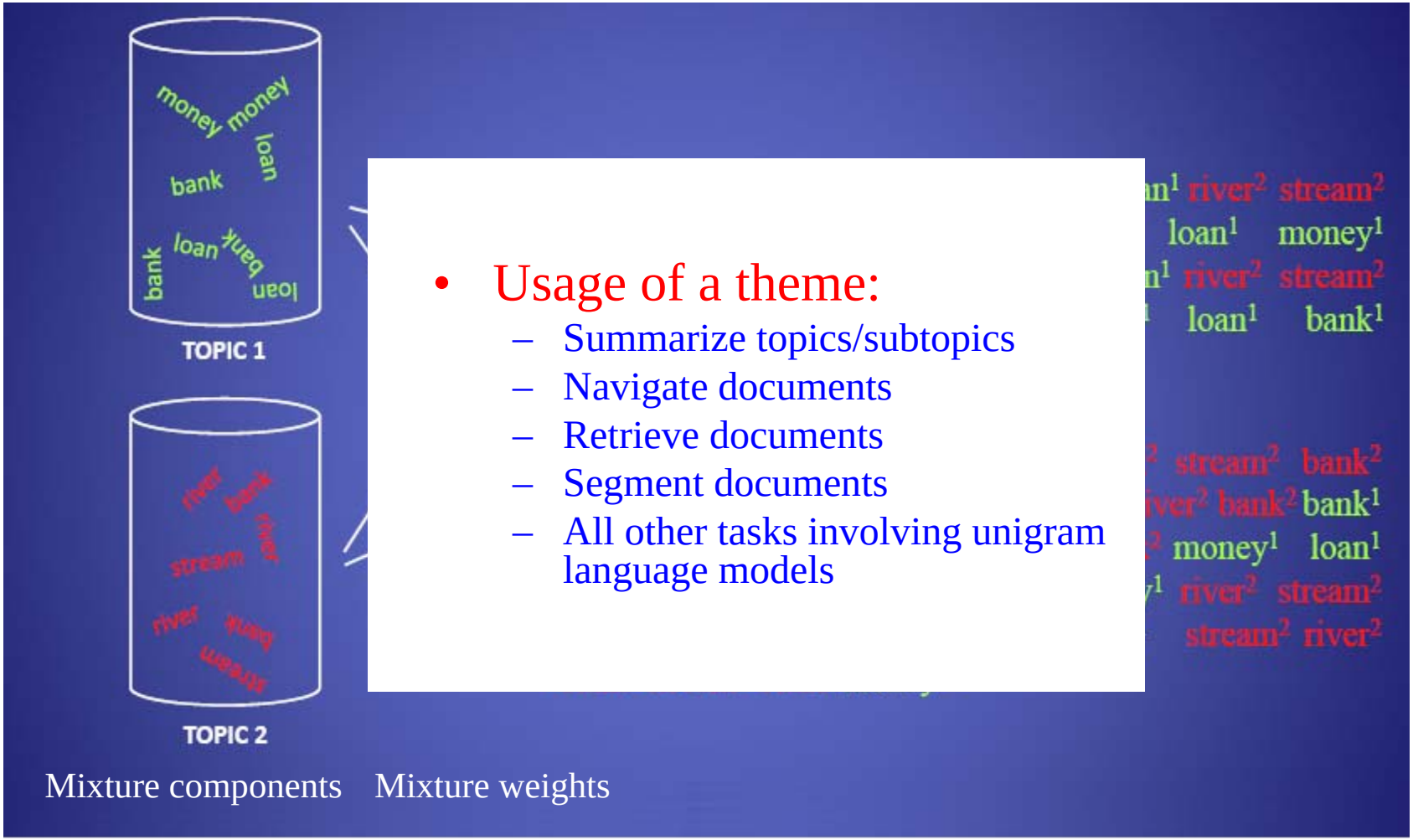
XU Shuo (徐硕)
E-mail: xush@istic.ac.cn
@中信所419会议室
2011-11-10

What is Topic?

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

What is Topic? (cont.)



- Usage of a theme:
  - Summarize topics/subtopics
  - Navigate documents
  - Retrieve documents
  - Segment documents
  - All other tasks involving unigram language models

What is pLSI?

- pLSI: probabilistic Latent Semantic Index

- pLSI is a generative model for generating the co-occurrence of documents $d \in \mathcal{D}=\{d_1,\ldots,d_M\}$ and terms $w \in \mathcal{W}=\{w_1,\ldots,w_N\}$, which associates latent variable $z \in \mathcal{Z}=\{z_1,\ldots,z_K\}$.

Assumptions in pLSI Model

- Observation pairs (d, w) are assumed to be generated independently
- Conditioned on the latent topic z, words w are generated independently of the specific document identity d.
- The words in a document are assumed to be exchangeable.
- The documents are assumed to be exchangeable.
- The latent topics z are assumed to be independent.

The Rules of Probability

- Sum rule

$$P(X) = \sum_{Y} P(X, Y)$$

- Product rule

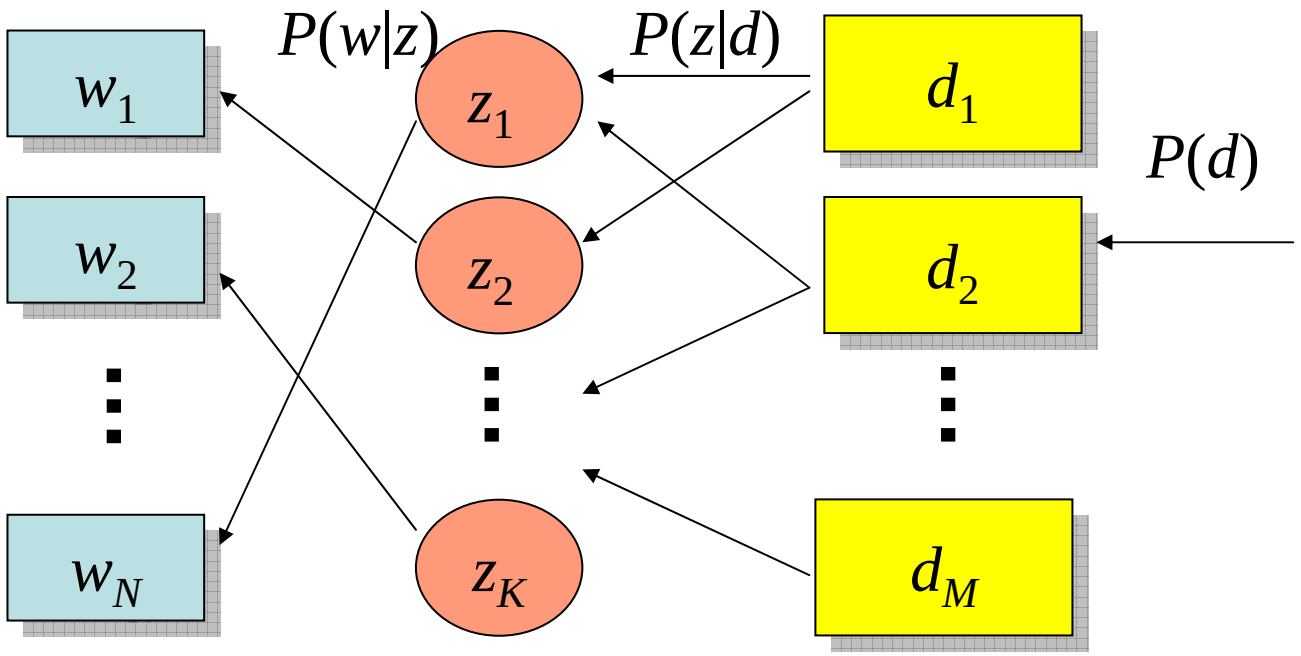
$$P(X, Y) = P(Y \mid X) P(X)$$

- Bayes rule

$$P(Y \mid X) = \frac{P(X, Y)}{P(X)} = \frac{P(X \mid Y) P(Y)}{P(X)}$$

pLSI Model

- The generative processing is:



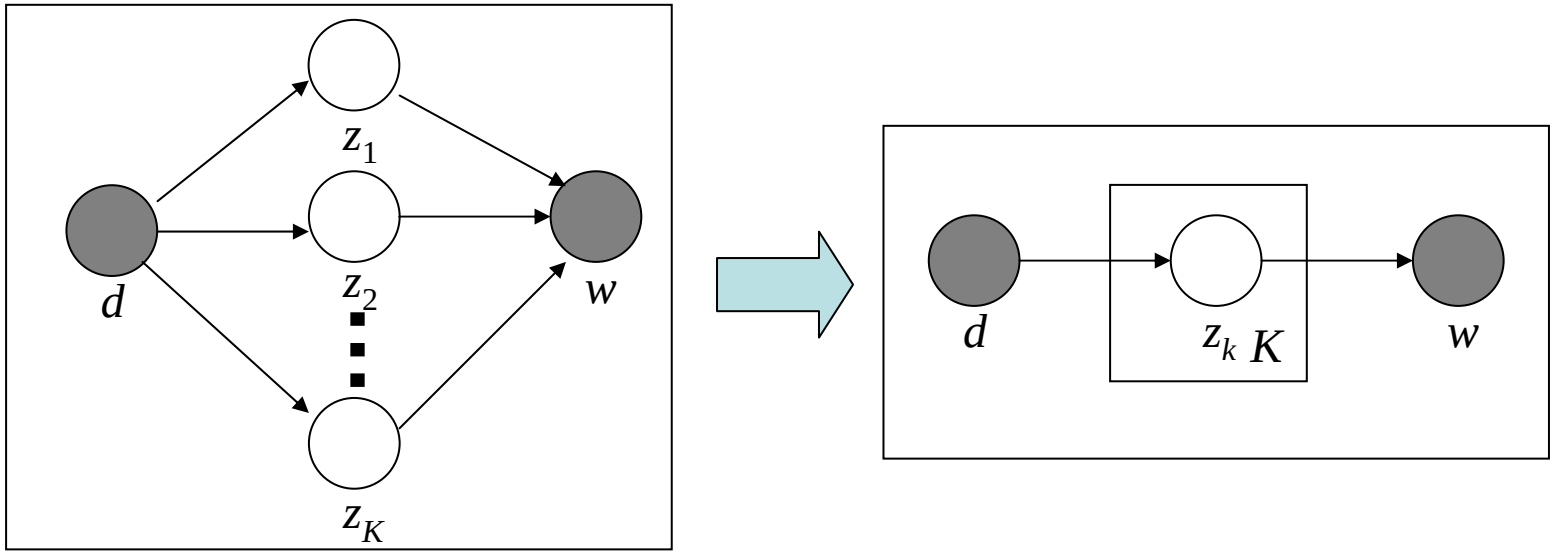
pLSI Model (cont.)

- Translate this process into a joint probability model:

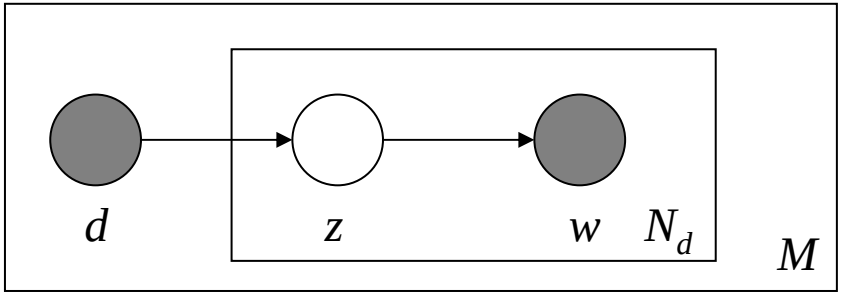
$$
\begin{aligned}
P(d,w) &\overset{\text{product rule}}{=} P(d)P(w\mid d) \\
&\overset{\text{sum rule}}{=} P(d)\sum_{z\in\mathcal{Z}} P(w,z\mid d) \\
&\overset{\text{product rule}}{=} P(d)\sum_{z\in\mathcal{Z}} P(z\mid d)P(w\mid z,d) \\
&\overset{\text{assumption 2}}{=} P(d)\sum_{z\in\mathcal{Z}} P(z\mid d)P(w\mid z)
\end{aligned}
$$

pLSI in Graphic Model

- Graphic Model is not a model, but a language.



$$P(d,w) = P(d)\sum_{z\in\mathcal{Z}} P(z\mid d)P(w\mid z)$$



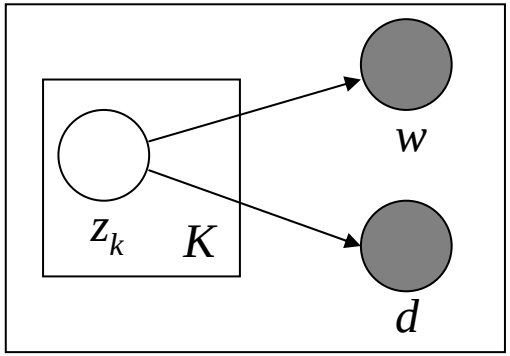
pLSI in Graphic Model (cont.)

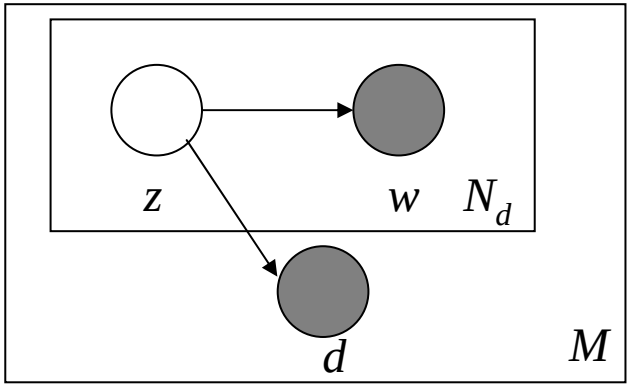
$$P(d,w) \overset{\text{sum rule}}{=} \sum_{z\in\mathcal{Z}} P(d,w,z)$$

$$\overset{\text{product rule}}{=} \sum_{z\in\mathcal{Z}} P(w\mid d,z)P(d,z)$$

$$\overset{\text{assumption 2}}{=} \sum_{z\in\mathcal{Z}} P(w\mid z)P(d,z)$$

$$\overset{\text{product rule}}{=} \sum_{z\in\mathcal{Z}} P(w\mid z)P(d\mid z)P(z)$$





Parameter Estimation

- How to estimate resulting parameters?
  - $P(z)$: K
  - $P(d|z)$: $M \times K$
  - $P(w|z)$: $N \times K$

$$\Theta = \left(\overbrace{P(z_1), \cdots, P(z_K)}^{P(z):K}, \overbrace{P(d_1 \mid z_1), \cdots, P(d_M \mid z_K)}^{P(d|z):M \times K}, \overbrace{P(w_1 \mid z_1), \cdots, P(w_N \mid z_K)}^{P(w|z):N \times K} \right)$$

- Maximum Likelihood Estimation:

$$\Theta = \arg\max_{\Theta} \mathcal{L}(\Theta) = \log\left(\prod_{i=1}^{M} \prod_{j=1}^{N_i} P(d_i, w^j) \right) = \log\left(\prod_{i=1}^{M} \prod_{j=1}^{N} P(d_i, w_j)^{n(d_i, w_j)} \right)$$

$$= \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} n(d, w) \log P(d, w)$$

$$= \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} n(d, w) \log\left(\sum_{z \in \mathcal{Z}} P(w \mid z) P(d \mid z) P(z) \right)$$

Parameter Estimation (cont.)

E-Step: $$P(z \mid d,w) = \frac{P(d,w,z)}{P(d,w)} = \boxed{\frac{P(w \mid z)P(d \mid z)P(z)}{\sum_{z \in Z} P(w \mid z)P(d \mid z)P(z)}}$$

M-Step:

$$P(w \mid z) = \frac{P(w,z)}{P(z)} = \frac{\sum_{d \in \mathcal{D}} P(d,w,z)}{\sum_{d' \in \mathcal{D}} \sum_{w' \in \mathcal{W}} P(d',w',z)} = \frac{\sum_{d \in \mathcal{D}} P(z \mid d,w)P(d,w)}{\sum_{d' \in \mathcal{D}} \sum_{w' \in \mathcal{W}} P(z \mid d',w')P(d',w')}$$

$$= \frac{\sum_{d \in \mathcal{D}} P(z \mid d,w)\frac{n(d,w)}{R}}{\sum_{d' \in \mathcal{D}} \sum_{w' \in \mathcal{W}} P(z \mid d',w')\frac{n(d',w')}{R}} = \boxed{\frac{\sum_{d \in \mathcal{D}} P(z \mid d,w)n(d,w)}{\sum_{d' \in \mathcal{D}} \sum_{w' \in \mathcal{W}} P(z \mid d',w')n(d',w')}}$$

$$P(d \mid z) = \boxed{\frac{\sum_{w \in \mathcal{W}} P(z \mid d,w)n(d,w)}{\sum_{d' \in \mathcal{D}} \sum_{w' \in \mathcal{W}} P(z \mid d',w')n(d',w')}}$$

Similar to $P(w|z)$

$$P(z) = \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} P(d,w)P(z \mid d,w) = \boxed{\frac{1}{R} \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} n(d,w)P(z \mid d,w)}$$

EM Algorithm

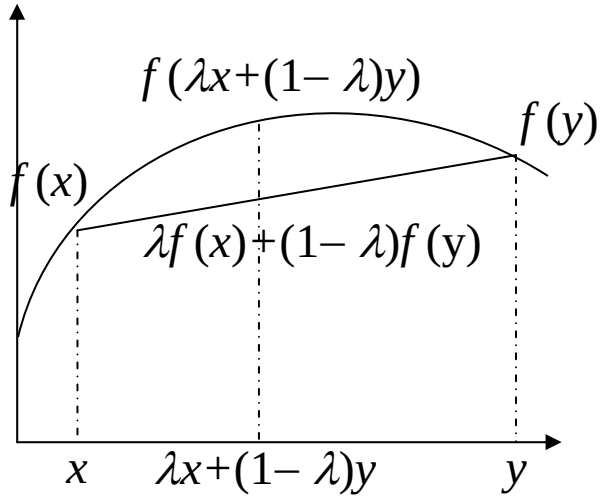
Definitions

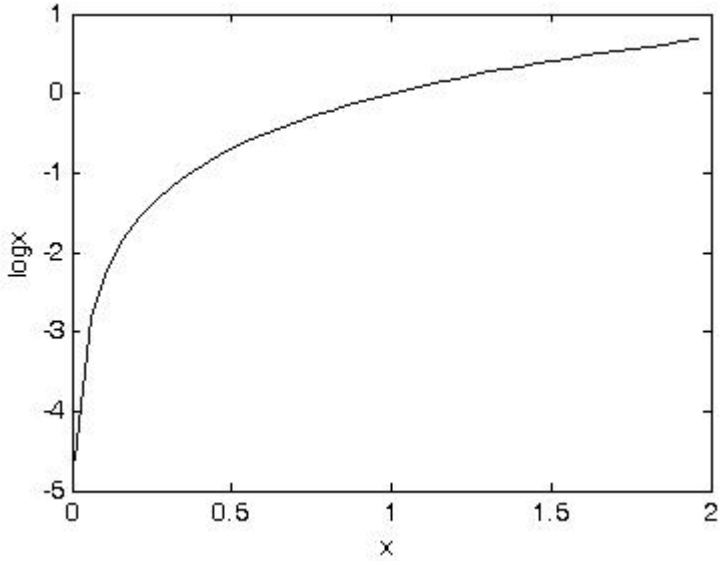
- The observed variables: X, e.g. $X = \{(d_1, w^1), \ldots, (d_M, w^{N_M})\}$
- The latent variables: Z, e.g. $Z = \{z_1, \ldots, z_K\}$
- The parameters needed to estimate: Θ, e.g. $P(z)$, $P(d|z)$, $P(w|z)$
- The distribution over the latent variables Z: $P(Z)$, e.g. $P(Z) = P(z_1) \times P(z_2) \times \ldots \times P(z_K)$
- Incomplete-data likelihood: $\mathcal{L}(\Theta) = P(X)$
- Complete-data likelihood: $\mathcal{L}_c(\Theta) = P(X, Z)$
- Our goal: $\Theta = \arg\max_{\Theta} \mathcal{L}(\Theta)$

Jensen's Inequality

- For any concave function $f(x)$, Jensen's inequality states that

$$f\left(\lambda x+(1-\lambda)x\right) \geq \lambda f(x)+(1-\lambda) f(y), \lambda \in [0,1]$$





Decomposition of $\mathcal{L}(\Theta)$

$$\log \mathcal{L}(\Theta) = \log P(X) = \log\left(\sum_Z P(X,Z) \right)$$

$$= \log\left(\sum_Z \frac{P(X,Z)}{P(Z)} P(Z) \right)$$

$$\geq \sum_Z P(Z) \log \frac{P(X,Z)}{P(Z)}$$

$$= \sum_Z P(Z) \log P(X,Z) - \sum_Z P(Z) \log P(Z)$$

$$\| \|$$

$$\mathbb{L}(P(Z),\Theta)$$

Decomposition of $\mathcal{L}(\Theta)$ (cont.)

$$\log \mathcal{L}(\Theta) = \log P(X) = \log \frac{P(X,Z)}{P(X \mid Z)} = \log P(X,Z) - \log P(X \mid Z)$$

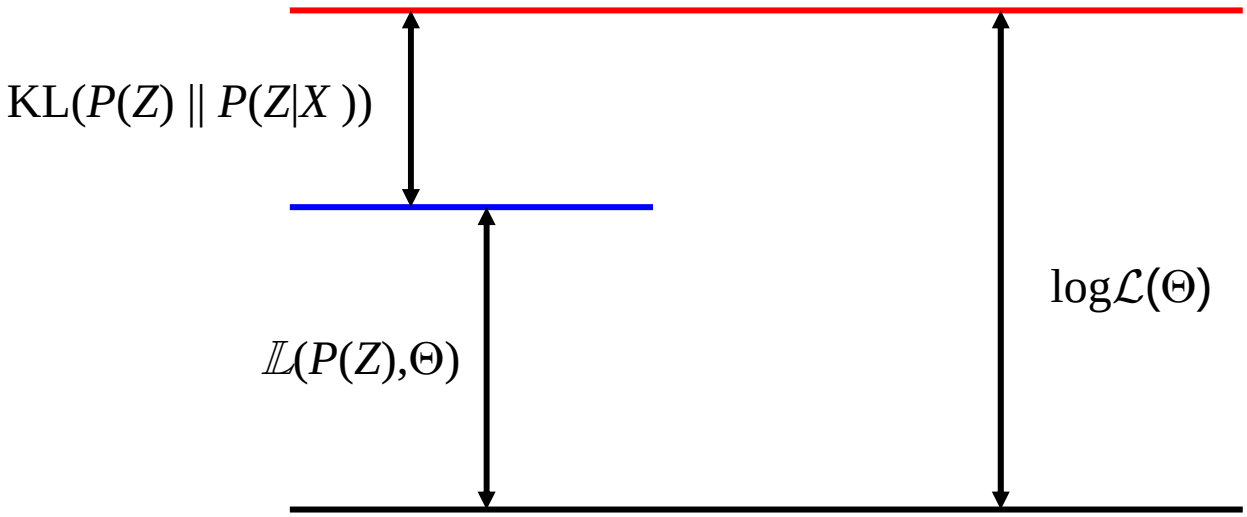
Multiplying two sides with $P(Z)$, and summing over Z yield

$$\log \mathcal{L}(\Theta) = \sum_Z P(Z) \log P(X,Z) - \sum_Z P(Z) \log P(X \mid Z)$$

Decomposition of $\mathcal{L}(\Theta)$ (cont.)

$$\log \mathcal{L}(\Theta) = \mathbb{L}(P(Z), \Theta) - \sum_Z P(Z) \log \frac{P(Z \mid X)}{P(Z)}$$

KL divergence ≥ 0 between $P(Z)$ and $P(Z|X)$ with equality iif $P(Z) = P(Z|X)$



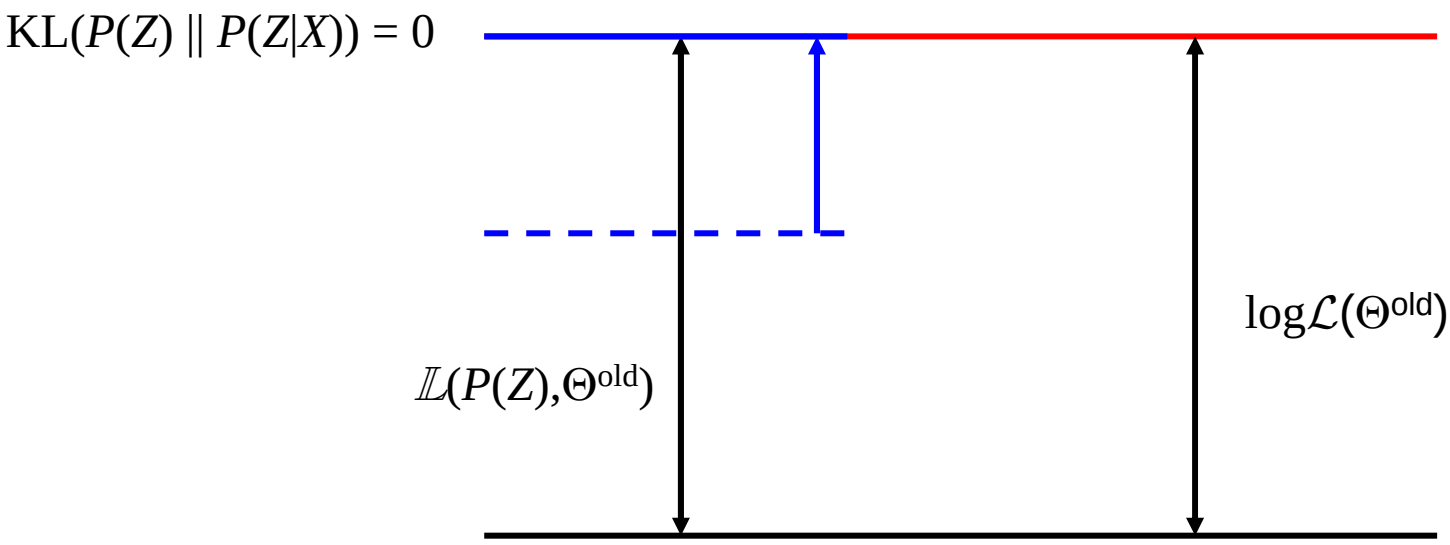
EM Algorithm

- STEP 1. Choose an initial setting for the parameters Θ^{old};
- STEP 2 (E step). Evaluate $P(Z|X; \Theta^{\text{old}})$;
- STEP 3 (M Step). Evaluate Θ^{new} given by

$$\Theta^{\text{new}} = \arg\max_{\Theta} \mathcal{Q}(\Theta; \Theta^{\text{old}}) = \sum_{Z} P(Z \mid X; \Theta^{\text{old}}) \log P(X, Z; \Theta)$$

- STEP 4. Check for convergence of either the log likelihood or the parameter values. If the convergence criterion is not satisfied, then let $\Theta^{\text{old}} \leftarrow \Theta^{\text{new}}$, and return to STEP 2.

EM Algorithm: Illustration



STEP 2 (E step). $P(Z) = P(Z|X; \Theta^{old})$;

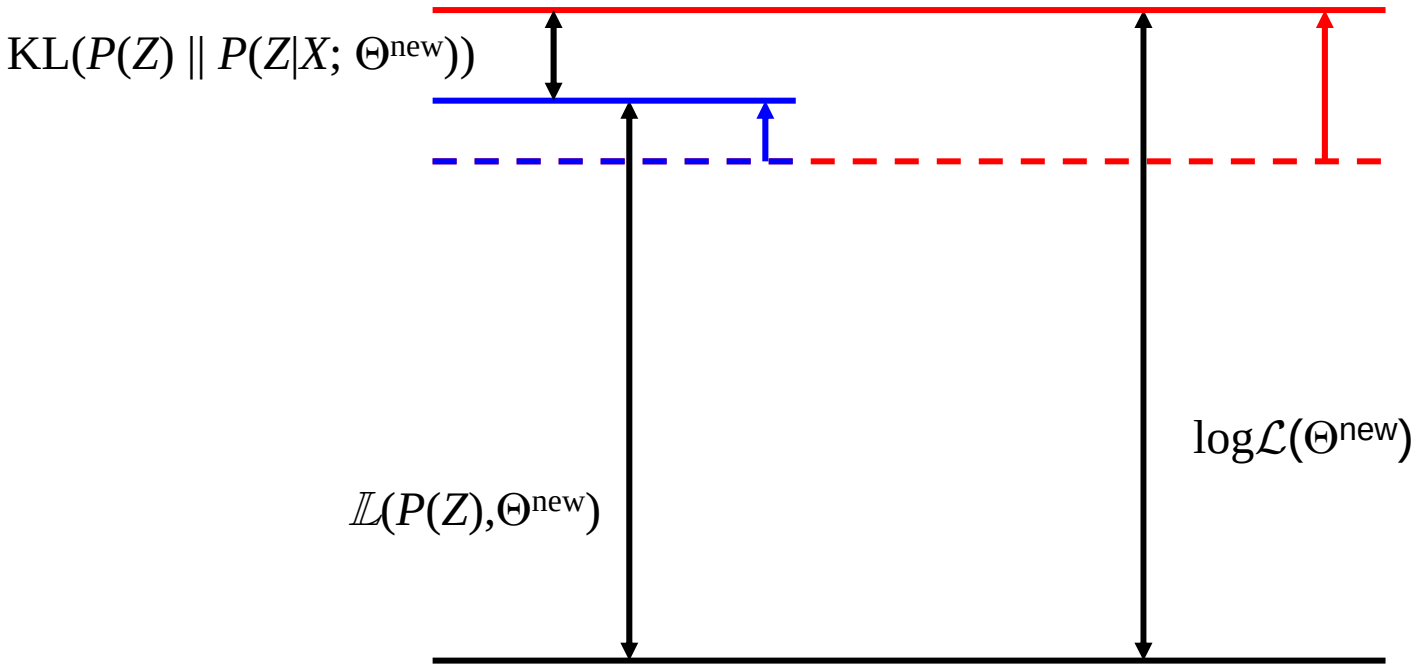
EM Algorithm: Illustration (cont.)

Substitute $P(Z) = P(Z|X; \Theta^{\text{old}})$ into $\mathbb{L}(P(Z), \Theta)$

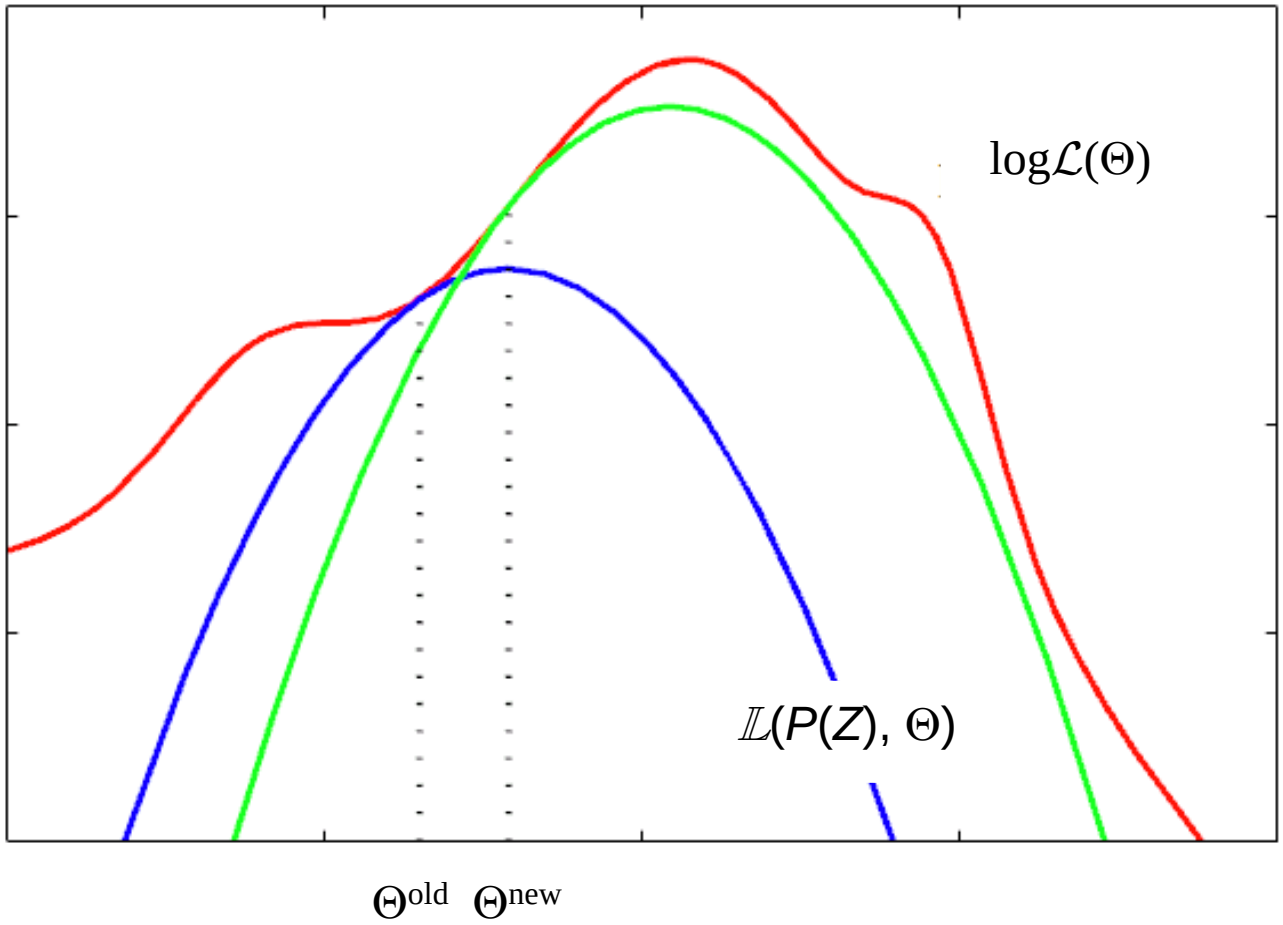
$$\log \mathbb{L}(\Theta;\Theta^{\text{old}}) = \sum_Z P(Z \mid X;\Theta^{\text{old}}) \log P(X,Z;\Theta) - \sum_Z P(Z \mid X;\Theta^{\text{old}}) \log P(Z \mid X;\Theta^{\text{old}})$$

$$= \mathcal{Q}(\Theta;\Theta^{\text{old}}) + \text{const}$$

STEP 3 (M Step). $\Theta^{\text{new}} = \arg\max_{\Theta} \mathcal{Q}(\Theta;\Theta^{\text{old}})$



EM Algorithm: Illustration (cont.)



Return to pLSI

Solve $P(w|z)$, $P(d|z)$ and $P(z)$

- Introduce Lagrange multiplier λ_z, α_z, and β for constraints $\Sigma_w P(w|z) = 1$, $\Sigma_d P(d|z) = 1$, and $\Sigma_z P(z) = 1$

$$\mathcal{F}(\Theta) = \mathcal{Q}(\Theta;\Theta^{\text{old}}) + \sum_{z\in\mathcal{Z}} \lambda_z \left(\sum_{w\in\mathcal{W}} P(w \mid z) - 1 \right) + \sum_{z\in\mathcal{Z}} \alpha_z \left(\sum_{d\in\mathcal{D}} P(d \mid z) - 1 \right) + \beta \left(\sum_{z\in\mathcal{Z}} P(z) - 1 \right)$$

- Solving the following equations:

$$\frac{\partial \mathcal{F}(\Theta)}{\partial P(w \mid z)} = \frac{\partial \mathcal{Q}(\Theta;\Theta^{\text{old}})}{\partial P(w \mid z)} + \lambda_z P(w \mid z) = 0$$

$$\frac{\partial \mathcal{F}(\Theta)}{\partial P(d \mid z)} = \frac{\partial \mathcal{Q}(\Theta;\Theta^{\text{old}})}{\partial P(d \mid z)} + \alpha_z P(d \mid z) = 0$$

$$\frac{\partial \mathcal{F}(\Theta)}{\partial P(z)} = \frac{\partial \mathcal{Q}(\Theta;\Theta^{\text{old}})}{\partial P(z)} + \beta P(z) = 0$$

Coding Design

- Variables:
  - **double**[][] *p_dz_n // p(d|z), |D|*|Z|*
  - **double**[][] *p_wz_n // p(w|z), |W|*|Z|*
  - **double**[] *p_z_n // p(z), |Z|*
- Running Processing:
  1. Read dataset from file

     ArrayList<DocWordPair> doc; // all the docs

     DocWordPair – (word_id, word_frequency_in_doc)
  2. Parameter Initialization

     Assign each elements of *p_dz_n, p_wz_n and p_z_n* with a random double value, satisfying $\sum_d p\_dz\_n=1$, $\sum_d p\_wz\_n=1$, and $\sum_d p\_z\_n=1$
  3. Estimation (Iterative processing)
     1. Update *p_dz_n, p_wz_n and p_z_n*
     2. Calculate Log-likelihood function to see where (|Log-likelihood – old_Log-likelihood| < threshold)
  4. Output *p_dz_n, p_wz_n and p_z_n*

Coding Design (cont.)

- **Update *p_dz_n***

```
For each doc d{
   For each word w included in d  {
       denominator = 0;
       nominator = new double[Z];
       For each topic z  {
          nominator[z] = p_dz_n[d][z]* p_wz_n[w][z]* p_z_n[z]
          denominator  +=nominator[z];
       } // end for each topic z
       For each topic z   {
           P_z_condition_d_w = nominator[j]/denominator;
           nominator_p_dz_n[d][z] += tf_wd*P_z_condition_d_w;
           denominator_p_dz_n[z] += tf_wd*P_z_condition_d_w;
       } // end for each topic z
    }// end for each word w included in d
}// end for each doc d

For each doc d {
     For each topic z            {
        p_dz_n_new[d][z] = nominator_p_dz_n[d][z]/ denominator_p_dz_n[z];
     } // end for each topic z
}// end for each doc d
```

Coding Design (cont.)

- **Update *p_wz_n***

```
For each doc d{
   For each word w included in d  {
       denominator = 0;
       nominator = new double[Z];
       For each topic z  {
          nominator[z] = p_dz_n[d][z]* p_wz_n[w][z]* p_z_n[z]
          denominator  +=nominator[z];
       } // end for each topic z
       For each topic z   {
          P_z_condition_d_w = nominator[j]/denominator;
          nominator_p_wz_n[w][z] += tf_wd*P_z_condition_d_w;
          denominator_p_wz_n[z] += tf_wd*P_z_condition_d_w;
       } // end for each topic z
    }// end for each word w included in d
}// end for each doc d

For each w {
     For each topic z            {
        p_wz_n_new[w][z] = nominator_p_wz_n[w][z]/ denominator_p_wz_n[z];
     } // end for each topic z
}// end for each doc d
```

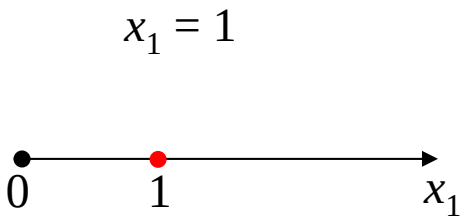
Coding Design (cont.)

- **Update *p_z_n***

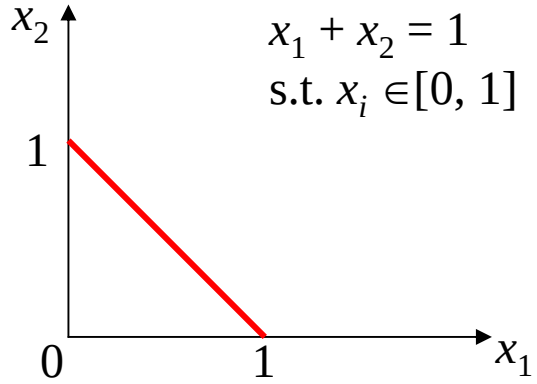
```
For each doc d{
   For each word w included in d  {
       denominator = 0;
       nominator = new double[Z];
       For each topic z  {
          nominator[z] = p_dz_n[d][z]* p_wz_n[w][z]* p_z_n[z]
          denominator  +=nominator[z];
       } // end for each topic z
       For each topic z   {
          P_z_condition_d_w = nominator[j]/denominator;
          nominator_p_z_n[z] += tf_wd*P_z_condition_d_w;
       } // end for each topic z
       denominator_p_z_n[z] += tf_wd;
    }// end for each word w included in d
}// end for each doc d

For each topic z{
       p_dz_n_new[d][j] = nominator_p_z_n[z]/ denominator_p_z_n;
} // end for each topic z
```

Interpretation

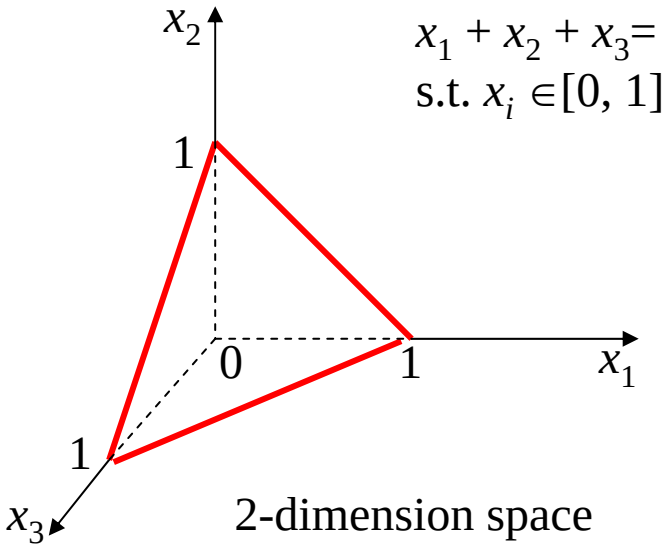
What is Simplex?



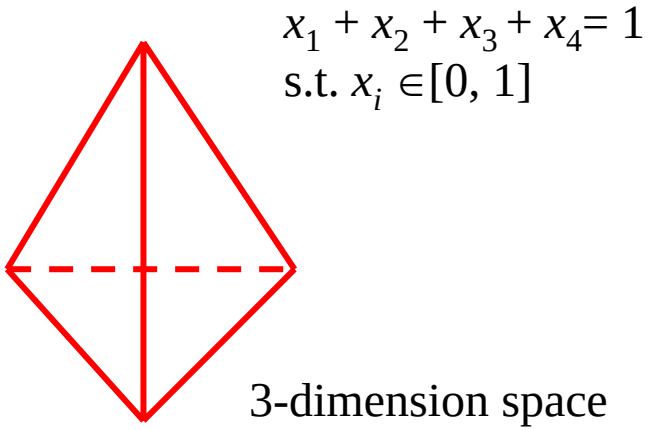
0-dimension space



1-dimension space

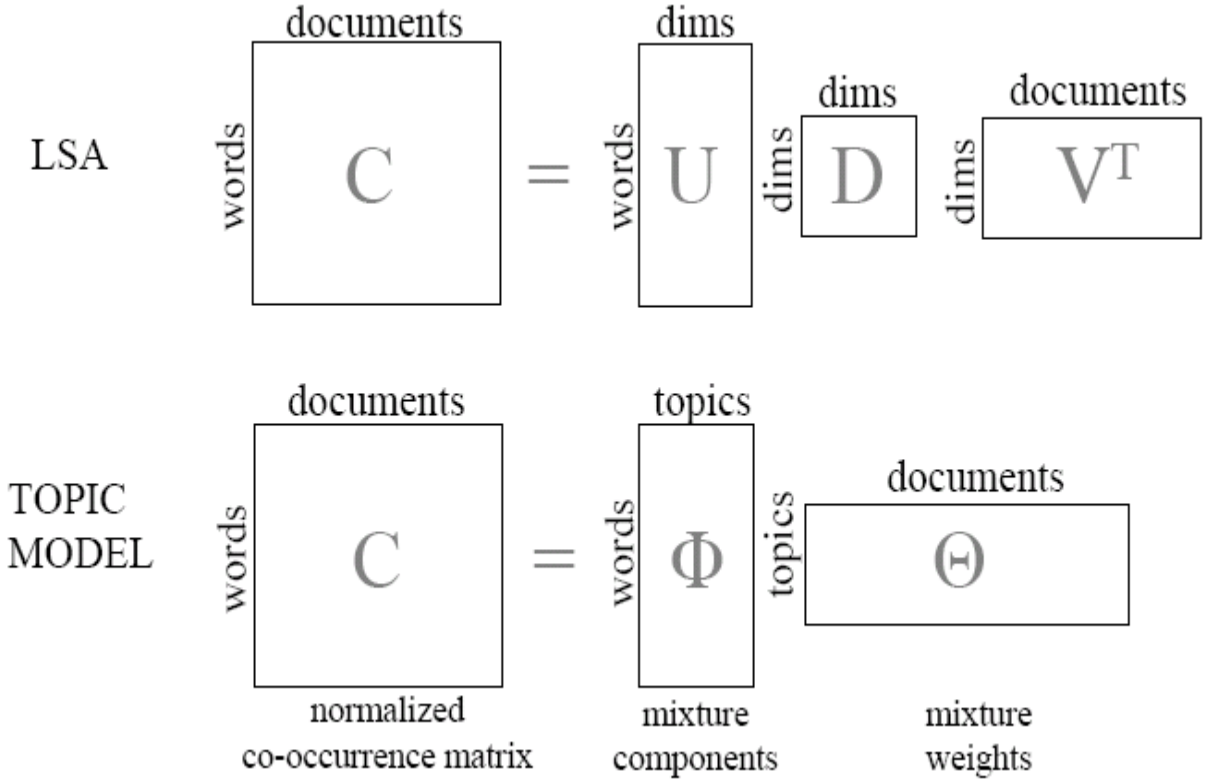


2-dimension space



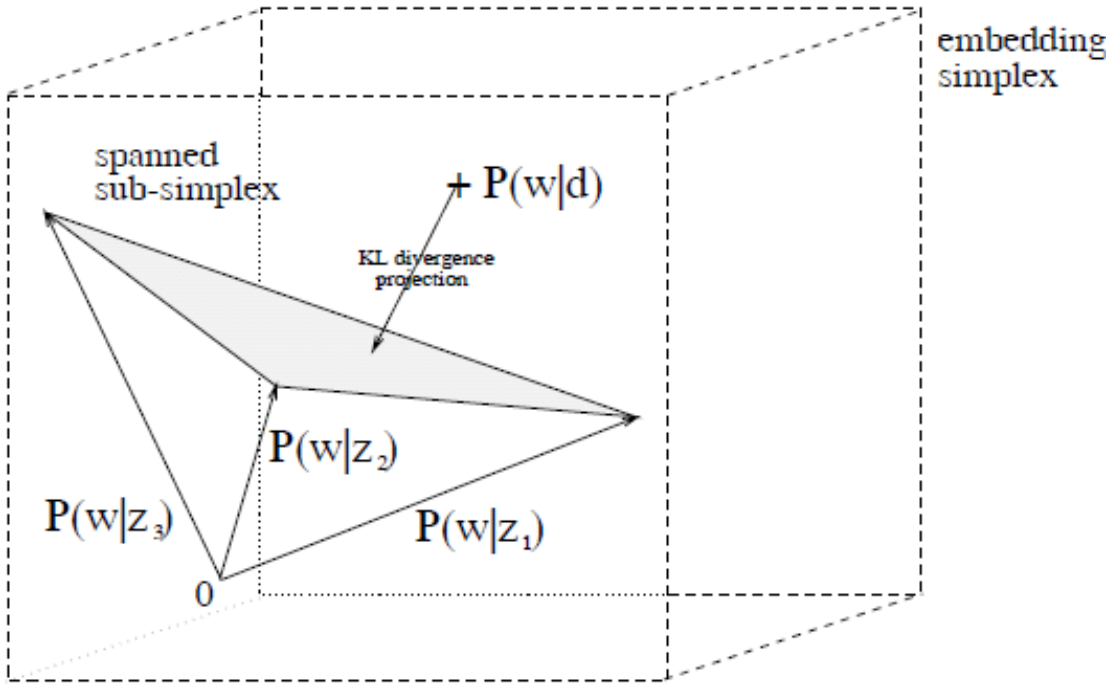
3-dimension space

Matrix Interpretation



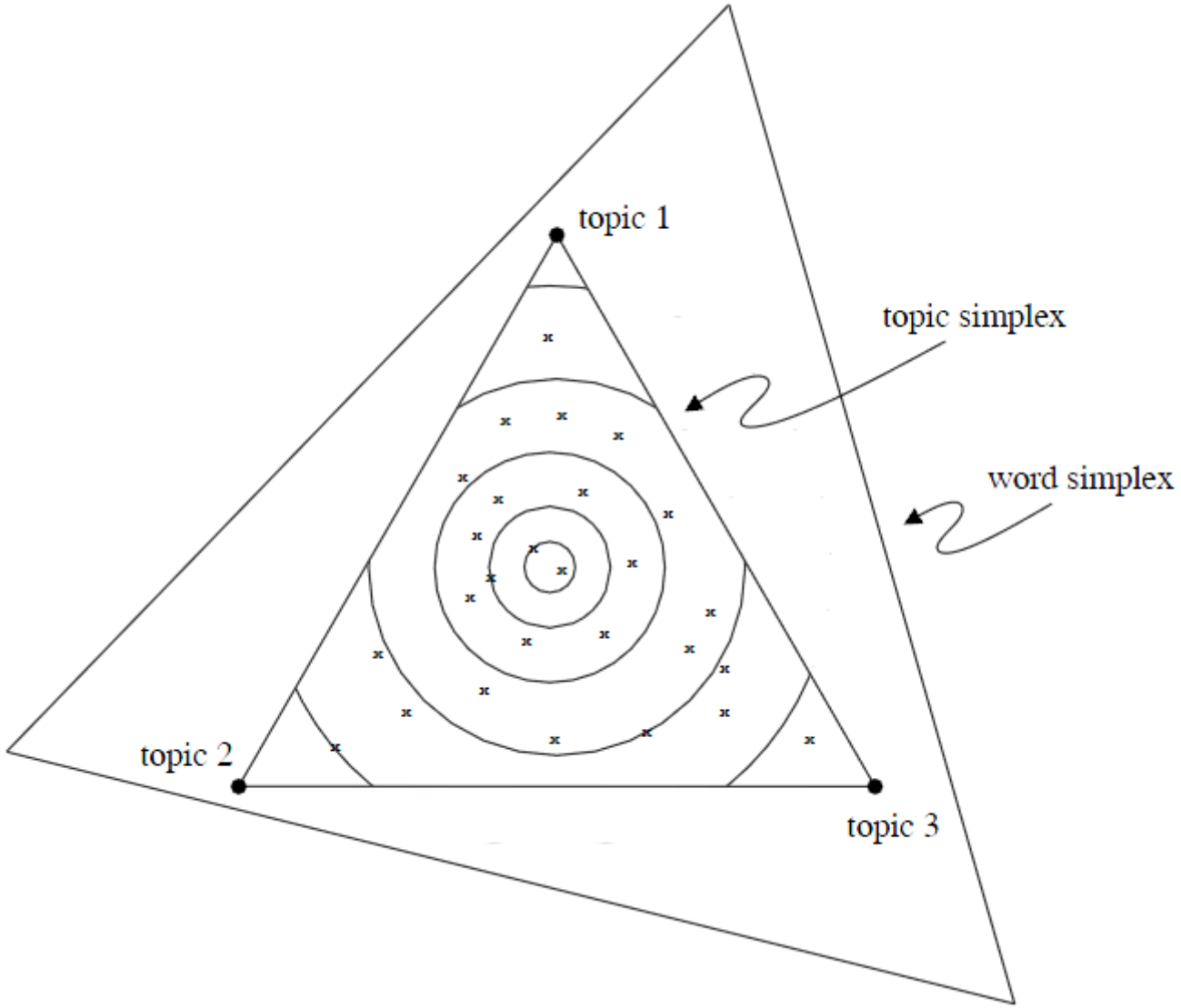
- In the topic model, the word-document co-occurrence matrix is split into two parts: a topic matrix Φ and a document matrix Θ. Note that the diagonal matrix D in LSA can be absorbed in the matrix U or V, making the similarity between the two representations even clearer.

Geometry Interpretation



$$P(w \mid d) = \sum_{z \in \mathcal{Z}} P(z \mid d) P(w \mid z)$$

Geometry Interpretation (cont.)



Advantages and Shortcomings

- Advantages
  - Significant compression in large collections
  - Capture synonymy and polysemy
  - Have solid statistical foundation
- Shortcomings
  - The number of parameters grows linearly with the size of the corpus
  - Folding-in problem: it is not clear how to assign probability to a document outside of the training set
  - Overfitting can occur even when tempering is used

References

- [1] Thomas Hofmann, 1999. Probabilistic Latent Semantic Indexing. SIGIR, pp. 50-57.
- [2] Liangjie Hong, 2010. A Tutorial on Probabilistic Latent Semantic Analysis. http://www.hongliangjie.com/notes/plsa.pdf
- [3] David M. Blei, Andrew Y. Ng and Michael I. Jordan, 2003. Latent Dirichlet Allocation. Journal of Machine Learning Research, Vol. 3, No. Jan, pp. 993-1022.
- [4] Christopher M. Bishop, 2006. Pattern Recognition and Machine Learning. Springer, New York.

Thanks

Thanks for your attention!

Q & A?