Matrix Factorization & Latent Semantic Analysis Review

Yize Li, Lanbo Zhang

Overview

- SVD in Latent Semantic Indexing
- Non-negative Matrix Factorization
- Probabilistic Latent Semantic Indexing

Vector Space Model

- A document: a vector in term space
- Vector computation: TF / TFIDF
- Similarity measure: angular cosine between query and documents.

$$\cos\theta_i = \frac{q * d_i}{|q| * |d_i|}$$

- Document vectors make up a term-document matrix.

Example

- 9 documents
- Terms in bold are in the dictionary.

c1: *Human* machine *interface* for Lab ABC *computer* applications
c2: A *survey* of *user* opinion of *computer system response time*
c3: The *EPS user interface* management *system*
c4: *System* and *human system* engineering testing of *EPS*
c5: Relation of *user*-perceived *response time* to error measurement

m1: The generation of random, binary, unordered *trees*
m2: The intersection *graph* of paths in *trees*
m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4: *Graph minors*: A *survey*

Term-Document Matrix (TF)

human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

Weakness of VSM

- Noise in term-document matrix
  - Synonyms
    - E.g. “car” & “automobile”.
    - Decrease recall
  - Polysems
    - E.g. “saturn”.
    - Decrease precision

Latent Semantic Indexing (LSI)

- A_{m*n}: term-document matrix
- Singular Value Decomposition (SVD)

$$\mathbf{A} = \mathbf{U}\mathbf{W}\mathbf{V}^{\mathrm{T}}$$

- Latent Semantic Indexing (LSI)

$$\mathbf{A}_k = \mathbf{U}_k \mathbf{W}_k \mathbf{V}_k^{\mathrm{T}}$$

What's really happening?

- Transformation of space
  - **Original: Term space**
    - Basis $B_1 = \{e_1, e_2, \ldots, e_m\}$, m is the term number in dictionary.
  - **New: Latent semantic space**
    - Basis $B_2 = \{u_1, u_2, \ldots, u_k\}$, k is the truncated dimension of document vector.

Thinking with LSI

- LSI aims to find
  - Meaning behind words
  - Topics in documents
- Difference between topics and words
  - Words – observable
  - Topics – latent
  - Topic space
  - Latent semantic space
  - Each basis vector u_i represents a topic

Evaluation of LSI

- Strength
  - Filter out noise(synonyms, polysems): dimension reduction considers only essential components of term-document matrix.
  - Reduces storage
- Weakness
  - Interpretation impossible: mixed signs
  - Orthogonal restriction on basis vector
  - Good truncation point k is hard to determine.

Non-negative Matrix Factorization

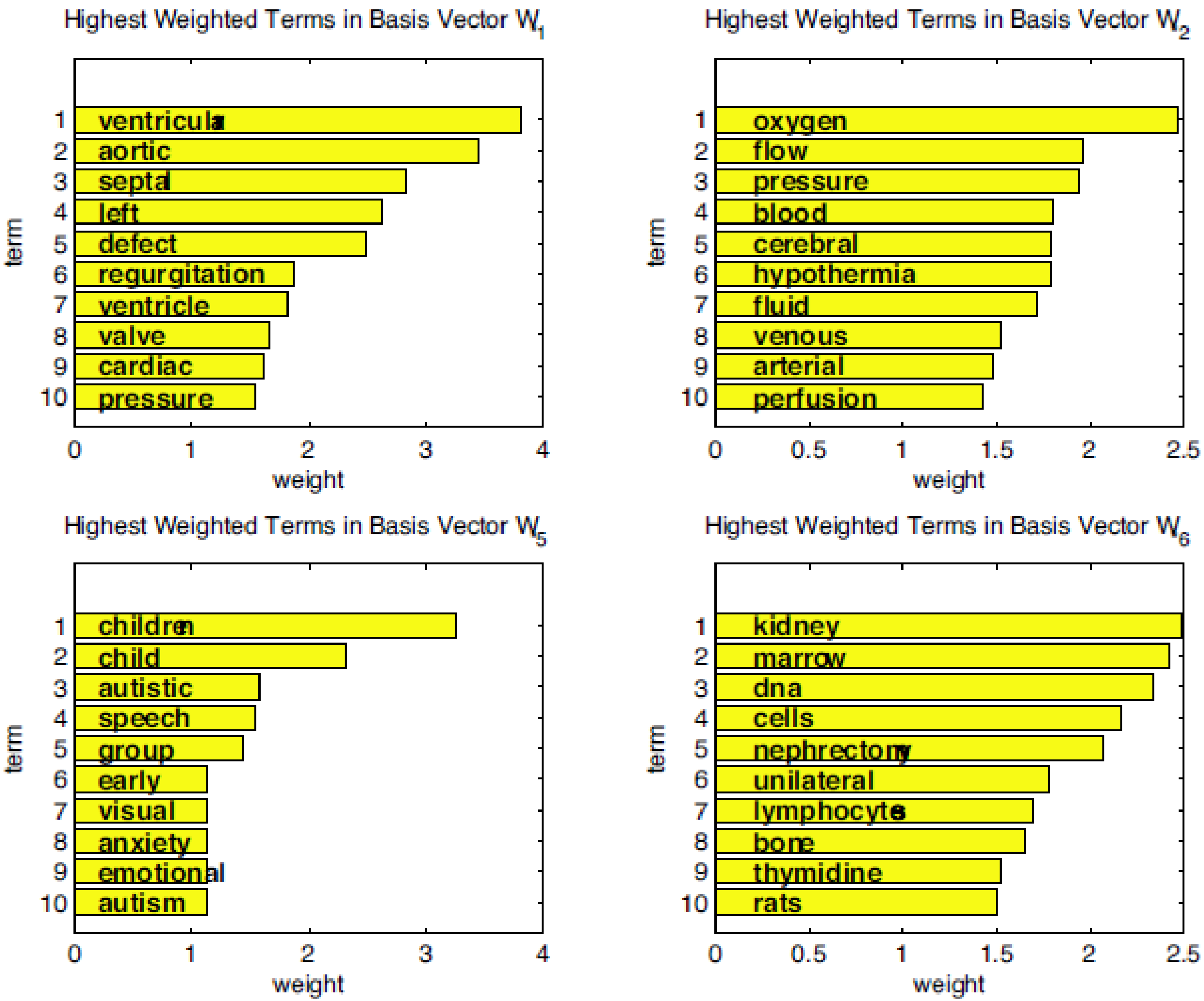
- Unlike SVD, we do matrix factorization as

$$\mathbf{A}_k = \mathbf{W}_k \mathbf{H}_k, \quad \mathbf{W}_k, \mathbf{H}_\mathrm{k} \geq \mathbf{0}$$

- Topic space
  - Dimension: k
  - Basis $b_3 = \{w_1, w_2, \ldots, w_k\}$

Properties of NMF

- No orthogonal restriction on basis vector
- Easy interpretation
  - Elements of W and H are all non-negative.
  - W_{ij} reflects how much basis vector w_j is related to term t_i
  - H_{ij} reflects how much document d_j points to the direction of basis vector w_i.



Highest Weighted Terms in Basis Vector W_1
ventricular
aortic
septal
left
defect
regurgitation
ventricle
valve
cardiac
pressure
term
weight
Highest Weighted Terms in Basis Vector W_2
oxygen
flow
pressure
blood
cerebral
hypothermia
fluid
venous
arterial
perfusion
term
weight
Highest Weighted Terms in Basis Vector W_5
children
child
autistic
speech
group
early
visual
anxiety
emotional
autism
term
weight
Highest Weighted Terms in Basis Vector W_6
kidney
marrow
dna
cells
nephrectomy
unilateral
lymphocytes
bone
thymidine
rats
term
weight

Computation of NMF

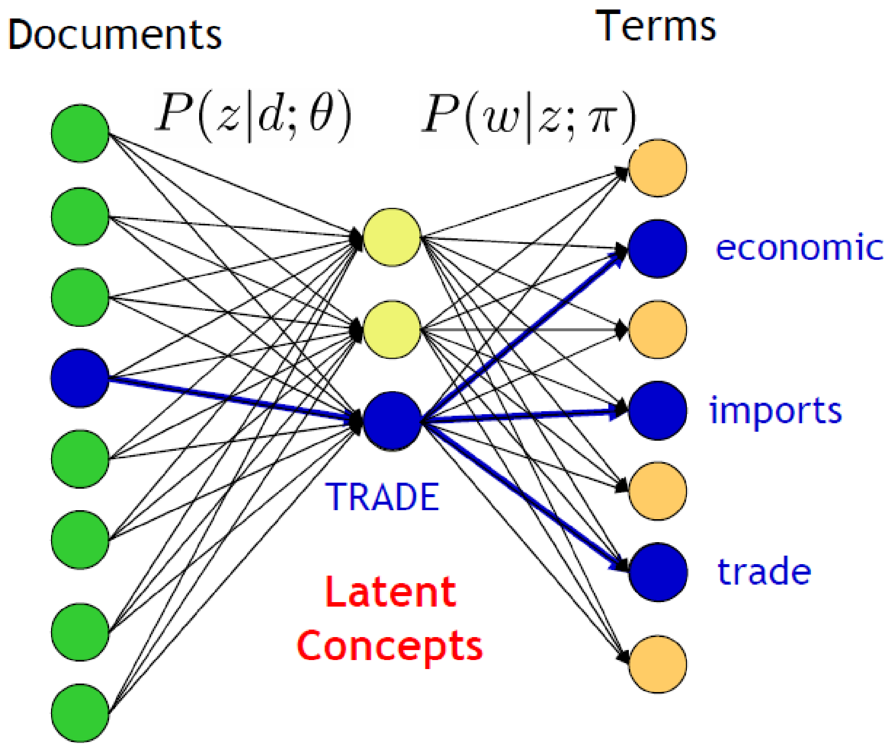
$$[\mathbf{W},\mathbf{H}] = \min \left\| \mathbf{A} - \mathbf{W}\mathbf{H} \right\|_F^2, \text{ s.t. } \mathbf{W},\mathbf{H} \geq \mathbf{0}$$

- Algorithms
  - Lee and Seung 2000
  - Berry etc. 2004

Evaluation of NMF

- Strength
  - Great interpretability
  - Improved Performance for document clustering comparing to LSI.
- Weakness
  - Factorization is not unique
  - Local minimum problem

pLSI: a probabilistic view of LSI



Why Latent Concepts?

Sparseness problem: terms not occurring in a document get zero probability

Concept expression probabilities are estimated based on all documents that are dealing with a concept

No prior knowledge about concepts required

Dimension reduction

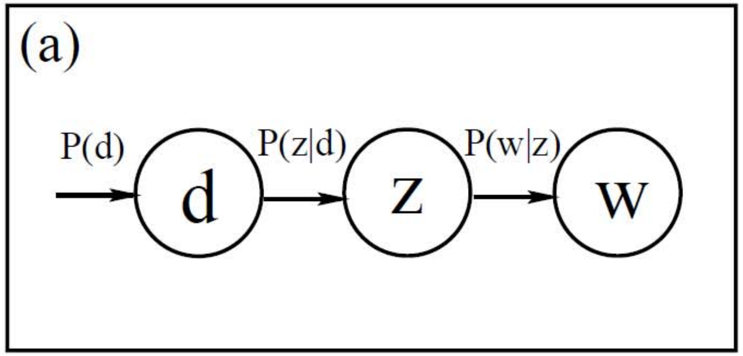
PLSA: Graphical model representation

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

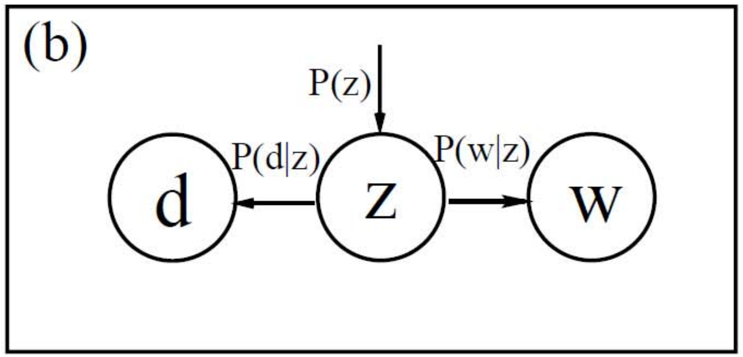
$$= \sum_{z\in Z} P(d)P(w \mid z)P(z \mid d) \quad \text{(a)}$$

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

$$= \sum_{z\in Z} P(z)P(w \mid z)P(d \mid z) \quad \text{(b)}$$



Asymmetric decomposition



Symmetric decomposition

pLSA via Likelihood Maximization

- Log-Likelihood

$$L(D,W) = \prod_{d,w} (\sum_z P(w \mid z) P(z \mid d))^{n(d,w)}$$

$$l = \sum_{d,w} n(d,w) \log(\sum_z P(w \mid z) P(z \mid d))$$

- Goal : maximize the log-likelihood with the constraints

$$\sum_w p(w|z_l) = 1, \sum_z p(z\ |d_j) = 1$$

KL Projection

KL divergence is a measure of difference between the empirical data distribution and the model

$$l = \sum_{d,w} n(d,w) \log(\sum_z P(w|z)P(z|d))$$

$$= \sum_d n(d)[\underline{\sum_w \frac{n(d,w)}{n(d)} \log P(w|d)} + \log P(d)]$$

Recall KL divergence is $D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$

$P = \hat{P}(w|d) = \frac{n(d,w)}{n(d)} \quad Q = P(w|d)$

Rewrite the underlined part: $-P \log \frac{1}{Q}$

PLSA via EM

- E-step: estimate posterior probabilities of latent variables, ("concepts")

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

Probability that the occurence of term w in document d can be "explained" by concept z

- M-step: parameter estimation based on expected statistics.

$$P(w \mid z) \propto \underbrace{\sum_{d} n(d, w) P(z \mid d, w)}_{}$$

how often is term *w* associated with concept *z*

$$P(d \mid z) \propto \underbrace{\sum_{w} n(d, w) P(z \mid d, w)}_{}$$

how often is document *d* associated with concept *z*

$$P(z) \propto \underbrace{\sum_{d, w} n(d, w) P(z \mid d, w)}_{}$$

probability of concept *z*

examples

10 most probable words in the respective latent classes (or factors)

“segment 1”	“segment 2”	“matrix 1”	“matrix 2”	“line 1”	“line 2”	“power 1”	power 2”
imag	speaker	robust	manufactur	constraint	alpha	POWER	load
SEGMENT	speech	MATRIX	cell	LINE	redshift	spectrum	memori
texture	recogni	eigenvalu	part	match	LINE	omega	vlsi
color	signal	uncertainti	MATRIX	locat	galaxi	mpc	POWER
tissue	train	plane	cellular	imag	quasar	hsup	systolic
brain	hmm	linear	famili	geometr	absorp	larg	input
slice	source	condition	design	impos	high	redshift	complex
cluster	speakerind.	perturb	machinepart	segment	ssup	galaxi	arrai
mri	SEGMENT	root	format	fundament	densiti	standard	present
volume	sound	suffici	group	recogn	veloc	model	implement

Most relevant latent classes for word “segment”

Most relevant latent classes for word “matrix”

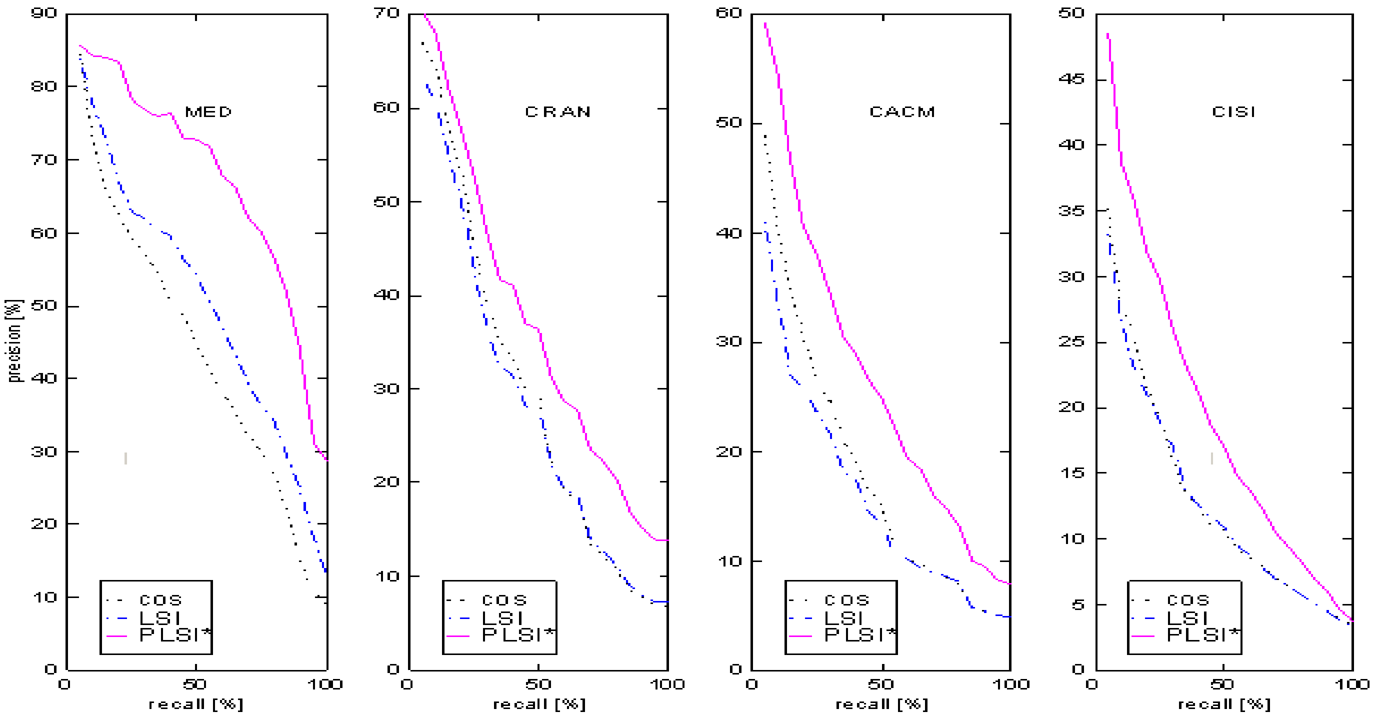
Most relevant latent classes for word “line”

Most relevant latent classes for word “power”

Model based on : 1568 documents on Clustering, Z=128

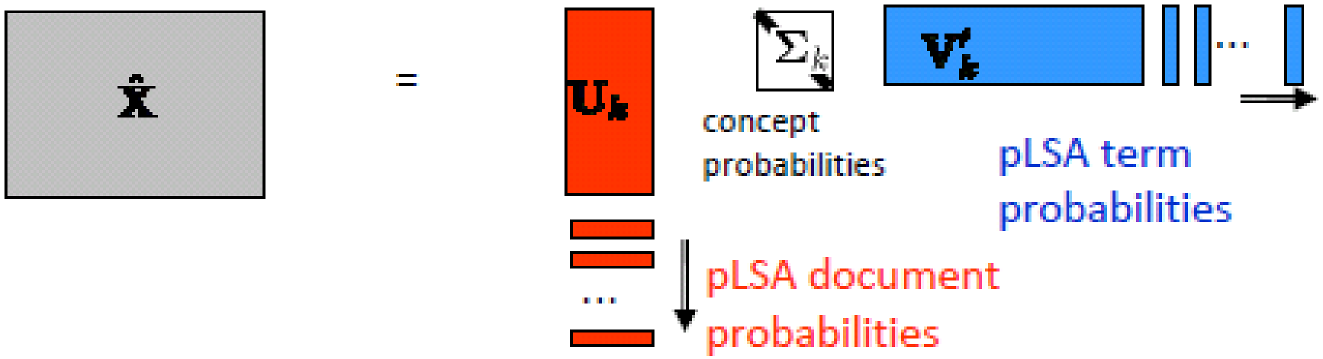
Performance comparison of a retrieval system: Three models, four document collection.

	MED		CRAN		CACM		CISI	
	prec.	impr.	prec.	impr.	prec.	impr.	prec.	impr.
cos+tf	44.3	-	29.9	-	17.9	-	12.7	-
LSI	51.7	+16.7	*28.7	-4.0	*16.0	-11.6	12.8	+0.8
PLSI	63.9	+44.2	35.1	+17.4	22.9	+27.9	18.8	+48.0



PLSA Mixture Decomposition vs. LSA/SVD

$$\hat{p}_{\mathrm{LSA}}(d, w) = \sum_z p(d|z)\, p(z)\, p(w|z)$$



PLSA vs. LSA

- Objective function: Frobenius norm vs. likelihood
- Non-negative
- Normalized
- There is no obvious interpretation of the directions in the LSA latent space; Multinomial word distribution in PLSA
- PLSA utilized statistical theory to determine the number of latent space dimension. LSA based on ad hoc heuristics

Relation between PLSA and NMP

- Any (local) maximum likelihood solution of PLSA is a solution of NMF with KL divergence
- KL divergence is a measure of the difference between the empirical distribution and the model
- Implications
  - Any problem which can be formulated with NMF, may be efficiently solved by PLSA

SVD in Collaborative Filtering

- Filling in missing values
  - Filling matrix using average value
  - EM algorithms

$$R = \begin{pmatrix} r_{11} & \cdots & r_{1M} \\ \vdots & \ddots & \vdots \\ r_{N1} & \cdots & r_{NM} \end{pmatrix} \approx \overbrace{\begin{pmatrix} w_{11} & \cdots & w_{1k} \\ \vdots & \ddots & \vdots \\ w_{N1} & \cdots & w_{Nk} \end{pmatrix}}^{k} \overbrace{\begin{pmatrix} v_{11} & \cdots & v_{1M} \\ \vdots & \ddots & \vdots \\ v_{k1} & \cdots & v_{NM} \end{pmatrix}}^{M}$$

Weighted -SVD

- Constant non-negative matrix $W \in \mathbb{R}_+^{n \times m}$.
- Weights the importance of each entry of the data matrix R

W

0	1	1	1	1	0
1	1	1	1	1	1
1	1	0	1	1	1
0	1	1	1	1	1

$\diamond$

R

	1	1	0	2	
7	0	1	3	0	2
5	0		0	1	2
	0	1	0	9	4

- Useful for masking missing entries of the matrix.
- Allows factorization to focus on certain pieces of the matrix.

NMF in Collaborative Filtering

- Objective: $Err(P,Q) = \sum_{(u,i)\in\kappa} (r_{ui} - p_u^T q_i)^2$

- Only deal with known values in R

- Can deal with large dataset

References

PLSA:

- T. Hofmann, Probabilistic Latent Semantic Analysis, Uncertainty in AI,1999
- T. Hofmann, Unsupervised Learning by Probabilistic Latent Semantic Analysis, Machine Learning Journal, 2000

EM for PCA/SVD:

- S. Roweis, EM Algorithm for PCA and SPCA, 1997
- S. Zhang, Using Singular Value Decomposition Approximation for Collaborative Filtering, CEC05, 2005

References(2)

NMF:

- I. Dhillon, Generalized Nonnegative Matrix Approximations with Bregman Divergences, NIPS2005
- D. Lee, Algorithms for Non-negative Matrix Factorization

PLSA vs NMF:

- E. Gaussier, Relation between PLSA and NMF and Implications, SIGIR'05
- C. Ding, On the equivalence between Non-negative Matrix Factorization and Probabilistic Latent Semantic Indexing, 2008

References(3)

Advanced topics:

- A. Singh, Relational Learning via Collective Matrix Factorization, KDD’08
- R. Bell, Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems, KDD07
- Y. Koren, Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model, KDD08
- C. Lippert, Relation Prediction in Multi-Relational Domains using Matrix Factorization, NIPS08