Recap: maximum likelihood estimation

- Data: a collection of words, $w_1, w_2, \dots, w_n$
- Model: multinomial distribution $\mathrm{p}(W)$ with parameters $\theta_i = p(w_i)$, i.e., unigram language model
- Maximum likelihood estimator: $\hat{\theta} = argmax_\theta p(W|\theta)$

$$p(W|\theta) = \binom{N}{c(w_1), \dots, c(w_N)} \prod_{i=1}^{N} \theta_i^{c(w_i)} \propto \prod_{i=1}^{N} \theta_i^{c(w_i)} \Rightarrow \log p(W|\theta) = \sum_{i=1}^{N} c(w_i) \log \theta_i$$

$$\Rightarrow L(W,\theta) = \sum_{i=1}^{N} c(w_i) \log \theta_i + \lambda \left(\sum_{i=1}^{N} \theta_i - 1 \right)$$

Using Lagrange multiplier approach, we'll tune $\boldsymbol{\theta_i}$ to maximize $\boldsymbol{L(W,\theta)}$

$$\Rightarrow \frac{\partial L}{\partial \theta_i} = \frac{c(w_i)}{\theta_i} + \lambda \quad \rightarrow \quad \theta_i = -\frac{c(w_i)}{\lambda}$$

Set partial derivatives to zero

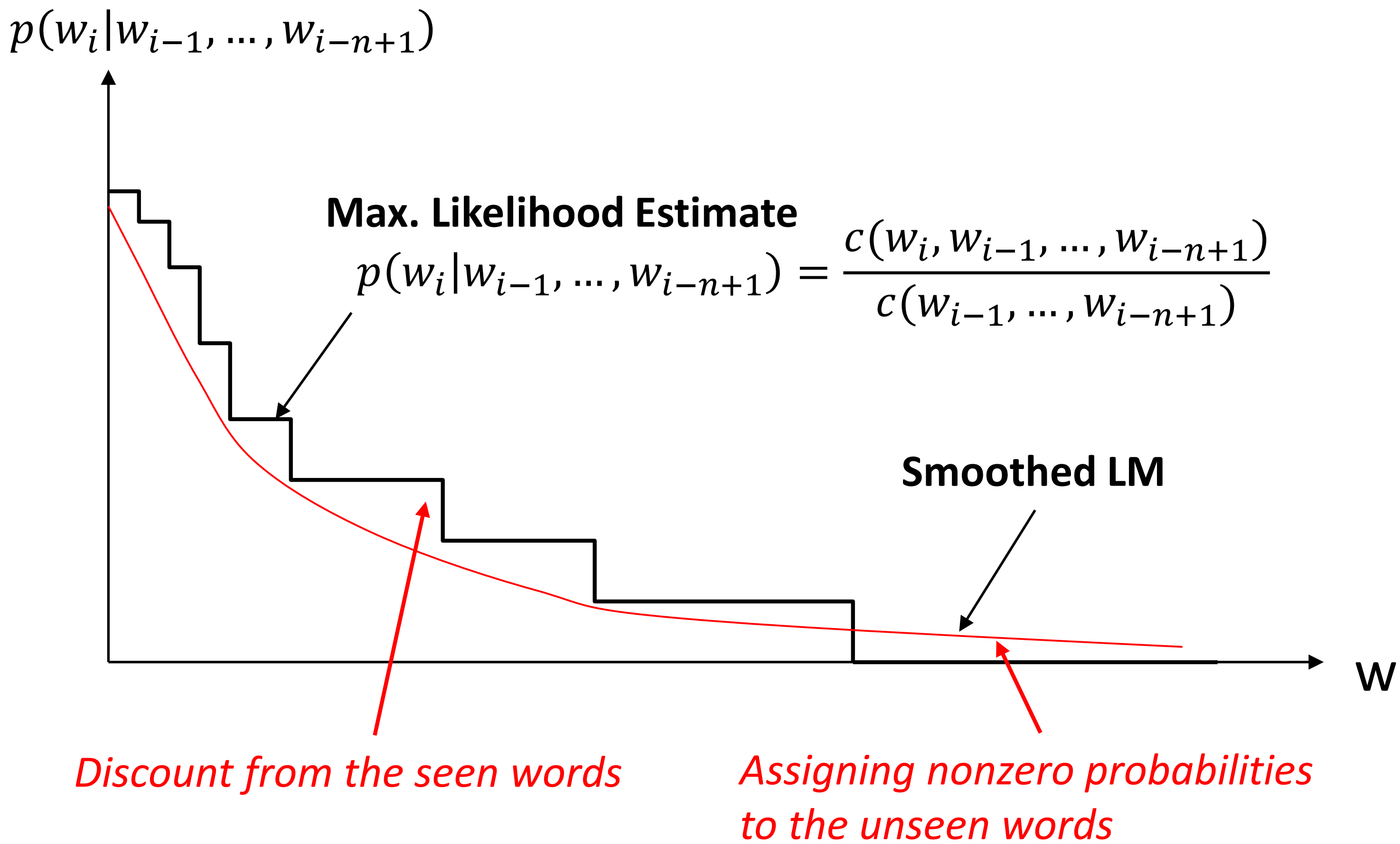
$\Rightarrow$ Since $\sum_{i=1}^{N} \theta_i = 1$ we have $\lambda = -\sum_{i=1}^{N} c(w_i)$

Requirement from probability

$$\Rightarrow \theta_i = \frac{c(w_i)}{\sum_{i=1}^{N} c(w_i)}$$

ML estimate

Recap: illustration of N-gram language model smoothing



Recap: perplexity

- The inverse of the likelihood of the test set as assigned by the language model, normalized by the number of words

$$PP(w_1, \dots, w_N) = \sqrt[N]{\frac{1}{\prod_{i=1}^{N} p(w_i | w_{i-1}, \dots, w_{i-n+1})}}$$

N-gram language model

Latent Semantic Analysis

Hongning Wang
CS@UVa

VS model in practice

- Document and query are represented by <u>term</u> vectors
  - Terms are not necessarily <u>orthogonal</u> to each other
    - Synonymy: car v.s. automobile
    - Polysemy: fly (action v.s. insect)

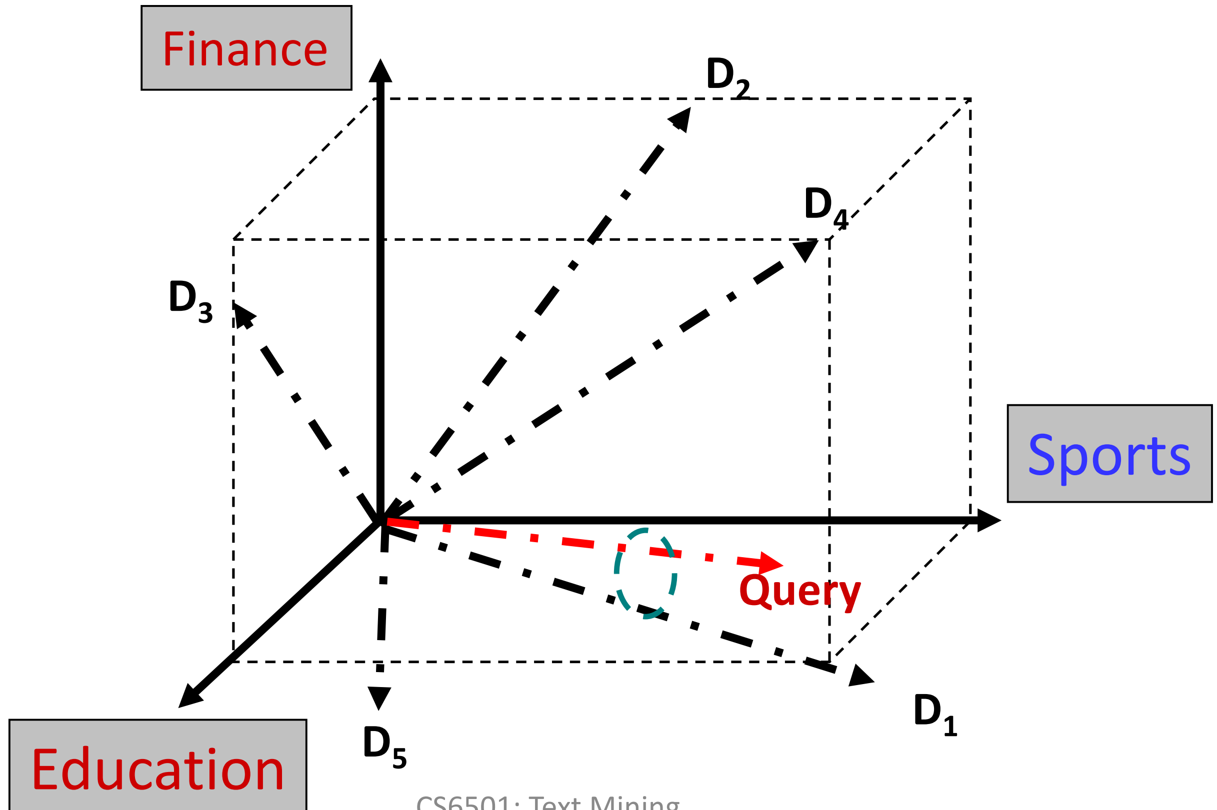
TABLE 1. Sample term by document matrix.[a]

	Access	Document	Retrieval	Information	Theory	Database	Indexing	Computer	REL	MATCH
Doc 1	x	x	x			x	x		R	
Doc 2				x*	x			x*		M
Doc 3			x	x*				x*	R	M

[a]Query: "IDF in *computer-based information look-up*"

Choosing basis for VS model

- A concept space is preferred
  - Semantic gap will be bridged



How to build such a space

- Automatic term expansion
  - Construction of thesaurus
    - WordNet
  - Clustering of words
- Word sense disambiguation
  - Dictionary-based
    - Relation between a pair of words should be similar as in text and dictionary's description
  - Explore word usage context

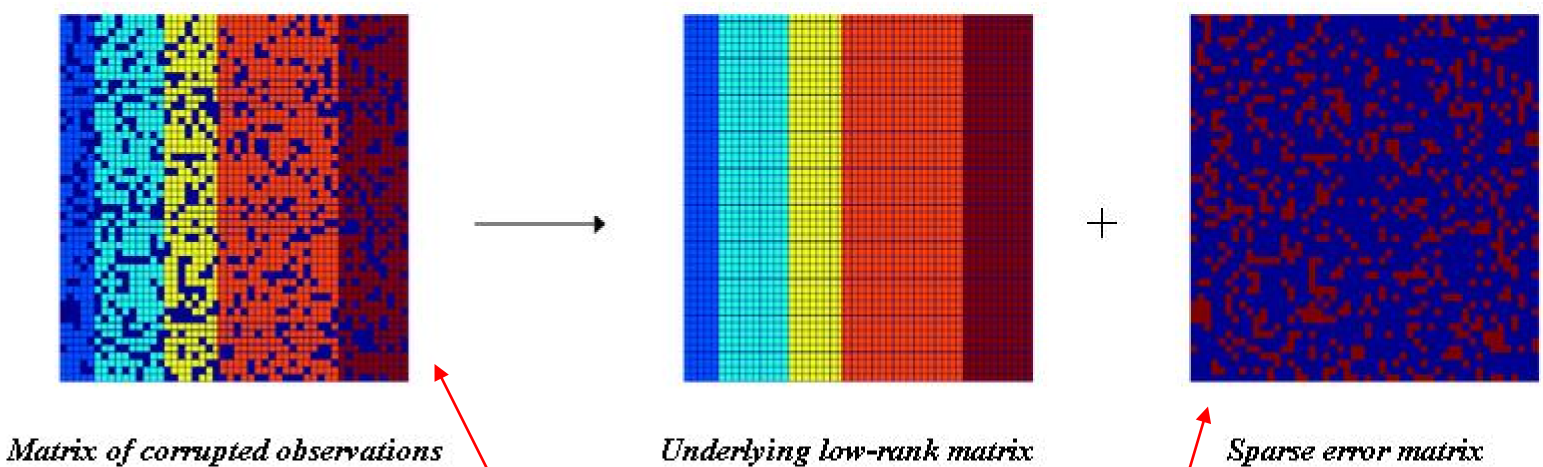
How to build such a space

- Latent Semantic Analysis
  - Assumption: there is some underlying latent semantic structure in the data that is partially obscured by the randomness of word choice with respect to text generation
  - It means: the observed term-document association data is contaminated by random noise

How to build such a space

- Solution
  - Low rank matrix approximation

*Imagine this is *true* concept-document matrix*



Imagine this is our observed term-document matrix

Random noise over the word selection in each document

Latent Semantic Analysis (LSA)

- Low rank approximation of term-document matrix $C_{M \times N}$
  - Goal: remove noise in the observed term-document association data
  - Solution: find a matrix with rank k which is closest to the original matrix in terms of Frobenius norm

$$\hat{Z} = \underset{Z|rank(Z)=k}{\operatorname{argmin}} \|C - Z\|_F$$

$$= \underset{Z|rank(Z)=k}{\operatorname{argmin}} \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} \left(C_{ij} - Z_{ij}\right)^2}$$

Basic concepts in linear algebra

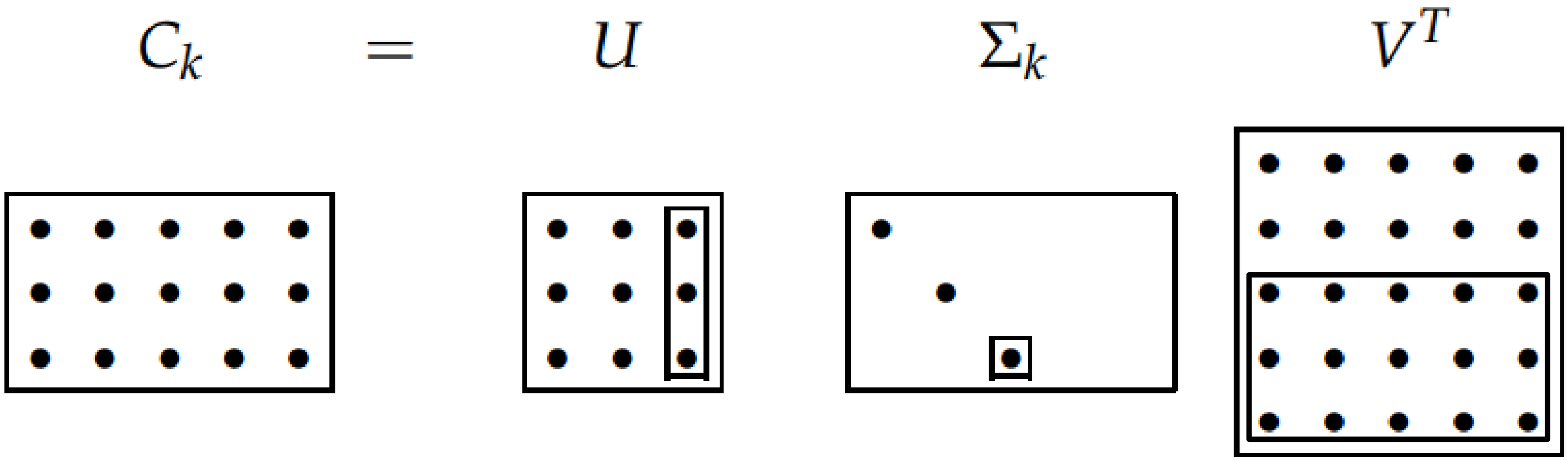
- Symmetric matrix
  - $C = C^T$
- Rank of a matrix
  - The number of linearly independent rows (columns) in a matrix $C_{M \times N}$
  - $rank(C_{M \times N}) \leq \min(M, N)$

Basic concepts in linear algebra

- Eigen system
  - For a square matrix $C_{M \times M}$
  - If $Cx = \lambda x$, x is called the right eigenvector of C and λ is the corresponding eigenvalue
- For a symmetric full-rank matrix $C_{M \times M}$
  - We have its eigen-decomposition as
    - $C = Q \Lambda Q^T$
    - where the columns of Q are the orthogonal and normalized eigenvectors of C and Λ is a diagonal matrix whose entries are the eigenvalues of C

Basic concepts in linear algebra

- Singular value decomposition (SVD)



  – We define $C^k_{M\times N} = U_{M\times k}\Sigma_{k\times k}V^T_{N\times k}$
    - where we place Σ_{ii} in a descending order and set $\Sigma_{ii} = \sqrt{\lambda_i}$ for $i \le k$, and $\Sigma_{ii} = 0$ for $i > k$

Latent Semantic Analysis (LSA)

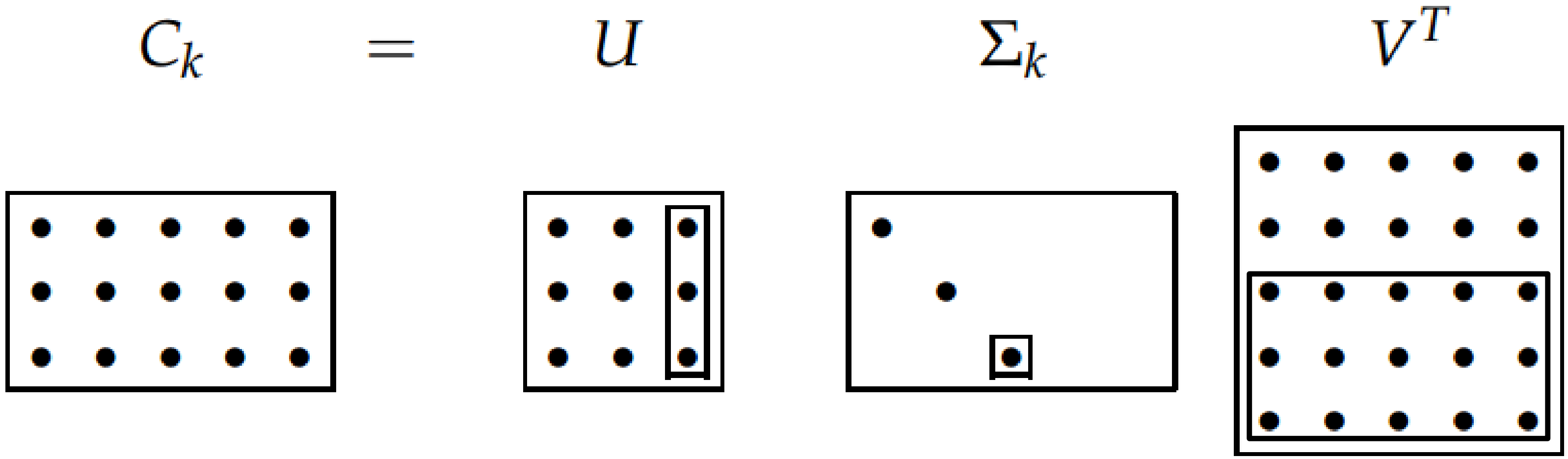
- Solve LSA by SVD

Map to a lower dimensional space

$$\hat{Z} = \underset{Z|rank(Z)=k}{\operatorname{argmin}} \|C - Z\|_F$$
$$= \underset{Z|rank(Z)=k}{\operatorname{argmin}} \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} \left(C_{ij} - Z_{ij}\right)^2}$$
$$= C_{M\times N}^{k}$$

  - Procedure of LSA
    1. Perform SVD on document-term adjacency matrix
    2. Construct $C_{M\times N}^{k}$ by only keeping the largest k singular values in Σ non-zero

Latent Semantic Analysis (LSA)

- 

  - $D = \left(U\Sigma V^T\right) \times \left(U\Sigma V^T\right)^T = U\Sigma^2 U^T$
    - Eigen-decomposition of document-document similarity matrix
    - d_i's new representation is then $(U\Sigma)_i$ in this system(space)
    - In the lower dimensional space, we will only use the first k elements in $(U\Sigma)_i$ to represent d_i
  - The same analysis applies to $T_{N\times N} = C_{M\times N}^T \times C_{M\times N}$

Geometric interpretation of LSA

- $C^k_{M\times N}(\mathrm{i},\mathrm{j})$ measures the relatedness between d_i and w_j in the k-dimensional space
- Therefore
  – As $C^k_{M\times N} = U_{M\times k}\Sigma_{k\times k}V^T_{N\times k}$
  – d_i is represented as $\left(U_{M\times k}\Sigma^{\frac{1}{2}}_{k\times k}\right)_i$
  – w_j is represented as $\left(V_{N\times k}\Sigma^{\frac{1}{2}}_{k\times k}\right)_j$

Latent Semantic Analysis (LSA)

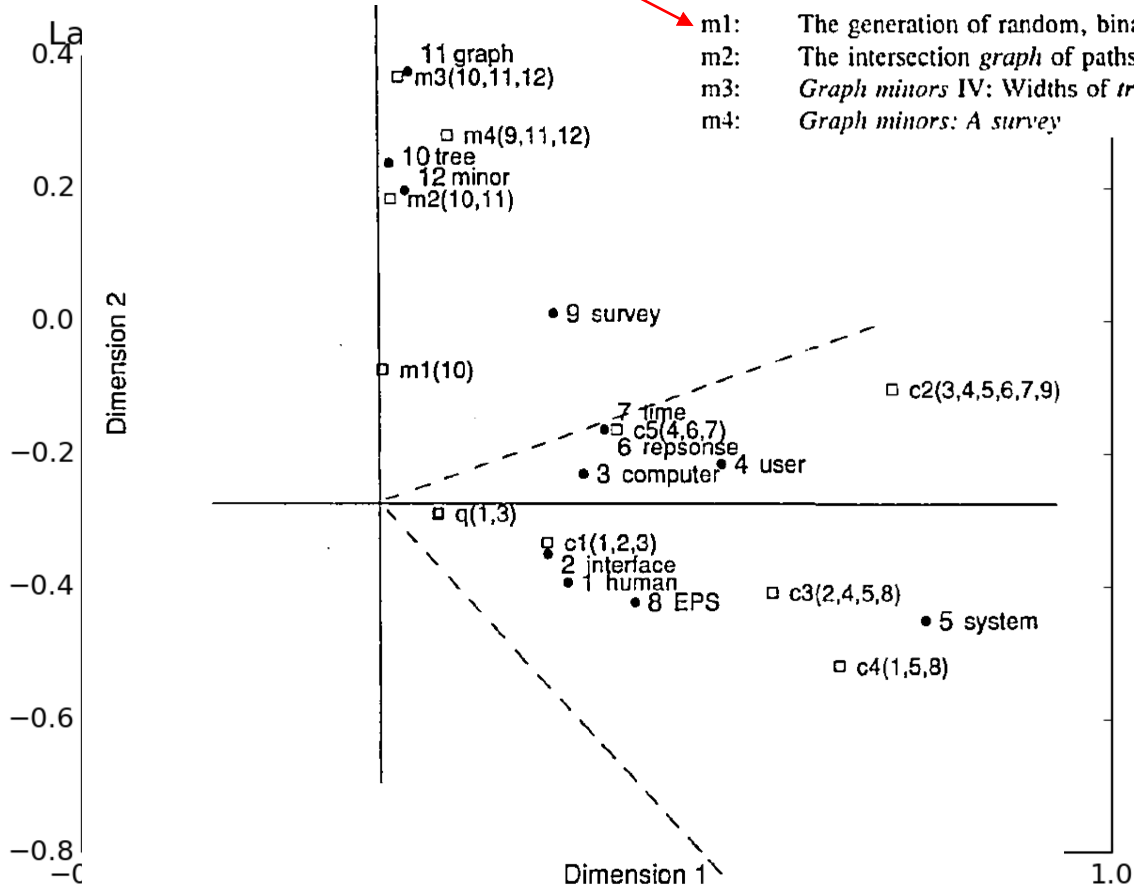
- Visualization

HCI

Graph theory

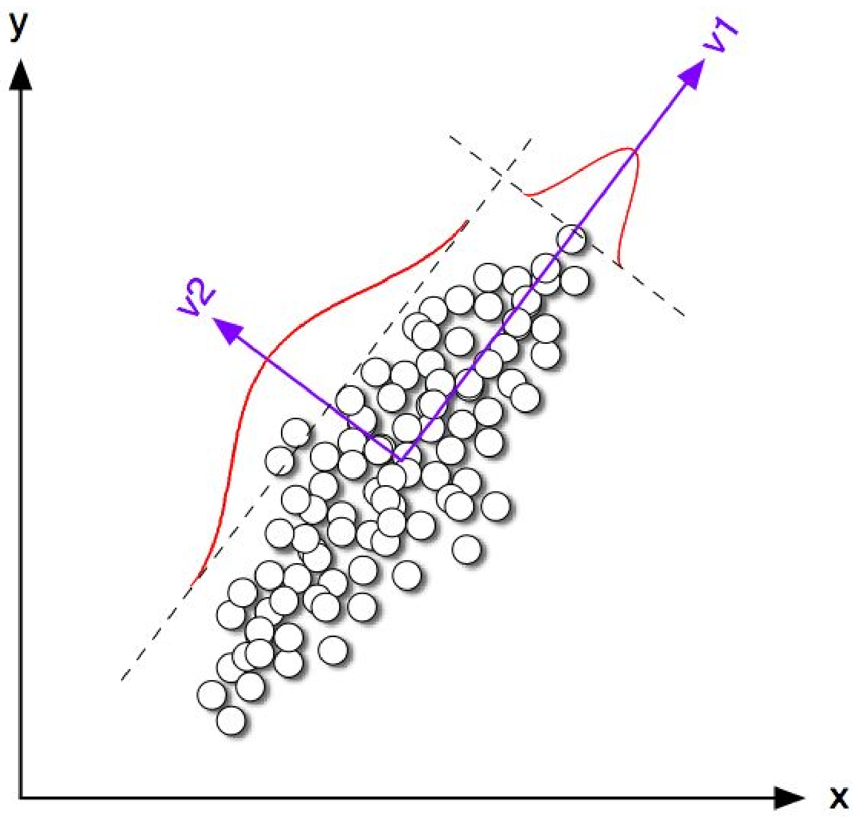
Titles

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*



What are those dimensions in LSA

- Principle component analysis



Latent Semantic Analysis (LSA)

- What we have achieved via LSA
  - Terms/documents that are closely associated are placed near one another in this new space
  - Terms that do not occur in a document may still close to it, if that is consistent with the major patterns of association in the data
  - A good choice of concept space for VS model!

LSA for retrieval

- Project queries into the new document space
  - $\tilde{q} = qV_{N \times k}\Sigma_{k \times k}^{-1}$
    - Treat query as a pseudo document of term vector
    - Cosine similarity between query and documents in this lower-dimensional space

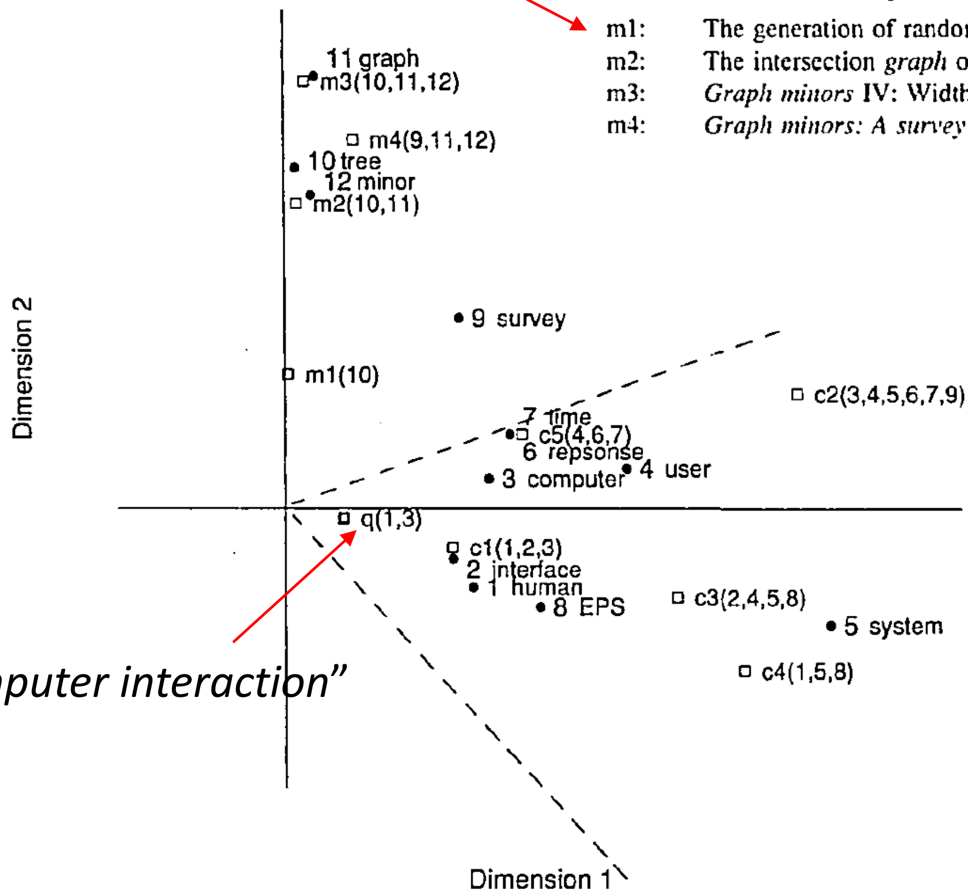
LSA for retrieval

HCI

Graph theory

Titles

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*



q: "*human computer interaction*"

Discussions

- Computationally expensive
  - Time complexity $O(MN^2)$
- Optimal choice of k
- Difficult to handle dynamic corpus
- Difficult to interpret the decomposition results

We will come back to this later!

LSA beyond text

- Collaborative filtering
  - User item matrix stores for each user the rating for the items

	i_1	i_2	i_3	i_4	i_5	…	i_M
u_1	2	0	3	2	5	…	1
u_2	0	4	0	0	0	…	5
u_3	0	2	0	0	0	…	4
u_4	1	0	4	2	4	…	2
…	…	…	…	…	…	…	…
u_K	2	…	4	…	4	…	1

Predicting unknown ratings

LSA beyond text

- Eigen face

LSA beyond text

- Cat from deep neuron network

One of the neurons in the artificial neural network, trained from still frames from unlabeled YouTube videos, learned to detect cats.

What you should know

- Assumptions in LSA
- Interpretation of LSA
  - Low rank matrix approximation
  - Eigen-decomposition of co-occurrence matrix for documents and terms

Today's reading

- Introduction to information retrieval
  - Chapter 13: Matrix decompositions and latent semantic indexing
- Deerwester, Scott C., et al. "Indexing by latent semantic analysis." *JAsIs* 41.6 (1990): 391-407.

Happy Lunar New Year!