CS276A
Text Retrieval and Mining

Lecture 15

Thanks to Thomas Hoffman, Brown University for sharing many of these slides.

Recap: Clustering 2

- Hierarchical clustering
  - Agglomerative clustering techniques
- Evaluation
- Term vs. document space clustering
- Multi-lingual docs
- Feature selection
- Labeling

Linear Algebra Background

Eigenvalues & Eigenvectors

- **Eigenvectors** (for a square $m \times m$ matrix $\mathbf{S}$)

$$\mathbf{S}\mathbf{v} = \lambda \mathbf{v}$$

(right) eigenvector $\mathbf{v} \in \mathbb{R}^m \neq \mathbf{0}$ — eigenvalue $\lambda \in \mathbb{R}$

Example

$$\begin{pmatrix} 6 & -2 \\ 4 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \end{pmatrix} = 2 \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

- How many eigenvalues are there at most?

$$\mathbf{S}\mathbf{v} = \lambda \mathbf{v} \iff (\mathbf{S} - \lambda \mathbf{I})\,\mathbf{v} = \mathbf{0}$$

only has a non-zero solution if $|\mathbf{S} - \lambda \mathbf{I}| = \mathbf{0}$

this is a m-th order equation in λ which can have **at most m distinct solutions** (roots of the characteristic polynomial) – can be complex even though **S** is real.

Matrix-vector multiplication

$$S = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

has eigenvalues 3, 2, 0 with corresponding eigenvectors

$$v_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad v_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad v_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

On each eigenvector, S acts as a multiple of the identity matrix: but as a different multiple on each.

Any vector (say $x = \begin{pmatrix} 2 \\ 4 \\ 6 \end{pmatrix}$) can be viewed as a combination of the eigenvectors: $x = 2v_1 + 4v_2 + 6v_3$

Matrix vector multiplication

- Thus a matrix-vector multiplication such as Sx (S, x as in the previous slide) can be rewritten in terms of the eigenvalues/vectors:

$$Sx = S(2v_1 + 4v_2 + 6v_3)$$

$$Sx = 2Sv_1 + 4Sv_2 + 6Sv_3 = 2\lambda_1 v_1 + 4\lambda_2 v_2 + 6\lambda_3 v_3$$

- Even though x is an arbitrary vector, the action of S on x is determined by the eigenvalues/vectors.
- Suggestion: the effect of "small" eigenvalues is small.

Eigenvalues & Eigenvectors

For symmetric matrices, eigenvectors for distinct eigenvalues are orthogonal

$$Sv_{\{1,2\}} = \lambda_{\{1,2\}} v_{\{1,2\}}, \text{ and } \lambda_1 \neq \lambda_2 \Rightarrow v_1 \bullet v_2 = 0$$

All eigenvalues of a real symmetric matrix are real.

$$\text{for complex } \lambda, \text{ if } |S - \lambda I| = 0 \text{ and } S = S^T \Rightarrow \lambda \in \Re$$

All eigenvalues of a positive semidefinite matrix are non-negative

$$\underbrace{\forall w \in \Re^n, w^T S w \geq 0}, \text{ then if } Sv = \lambda v \Rightarrow \lambda \geq 0$$

Example

- Let $S = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ ← Real, symmetric.
- Then $S - \lambda I = \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix} \Rightarrow (2-\lambda)^2 - 1 = 0.$
- The eigenvalues are 1 and 3 (nonnegative, real).
- The eigenvectors are orthogonal (and real):

$$\begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Plug in these values and solve for eigenvectors.

Eigen/diagonal Decomposition

- Let $\mathbf{S} \in \mathbb{R}^{m \times m}$ be a **square** matrix with ***m* linearly independent eigenvectors** (a "non-defective" matrix)
- **Theorem**: Exists an **eigen decomposition**

$$\mathbf{S} = \mathbf{U \Lambda U^{-1}}$$

diagonal

Unique for distinct eigen-values

  - (cf. matrix diagonalization theorem)
- Columns of ***U*** are **eigenvectors** of ***S***
- Diagonal elements of $\mathbf{\Lambda}$ are **eigenvalues** of $\mathbf{S}$

$$\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_m), \quad \lambda_i \geq \lambda_{i+1}$$

Diagonal decomposition: why/how

Let ***U*** have the eigenvectors as columns: $U = \begin{bmatrix} v_1 & \ldots & v_n \end{bmatrix}$

Then, ***SU*** can be written

$$SU = S\begin{bmatrix} v_1 & \ldots & v_n \end{bmatrix} = \begin{bmatrix} \lambda_1 v_1 & \ldots & \lambda_n v_n \end{bmatrix} = \begin{bmatrix} v_1 & \ldots & v_n \end{bmatrix}\begin{bmatrix} \lambda_1 & & \\ & \ldots & \\ & & \lambda_n \end{bmatrix}$$

Thus ***SU=UΛ***, or ***U⁻¹SU=Λ***

And ***S=UΛU⁻¹***.

Diagonal decomposition - example

Recall $S = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}; \lambda_1 = 1, \lambda_2 = 3.$

The eigenvectors $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$ and $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ form $U = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$

Inverting, we have $U^{-1} = \begin{bmatrix} 1/2 & -1/2 \\ 1/2 & 1/2 \end{bmatrix}$ ← Recall $UU^{-1} = 1$.

Then, ***S=UΛU⁻¹*** = $\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}\begin{bmatrix} 1/2 & -1/2 \\ 1/2 & 1/2 \end{bmatrix}$

Example continued

Let's divide ***U*** (and multiply ***U⁻¹***) by $\sqrt{2}$

Then, ***S***= $\begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}\begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$

Q ***Λ*** ***($Q^{-1}= Q^T$)***

Why? Stay tuned …

Symmetric Eigen Decomposition

- If $\mathbf{S} \in \mathbb{R}^{m \times m}$ is a **symmetric** matrix:
- **Theorem**: Exists a (unique) **eigen decomposition** $S = Q \Lambda Q^T$
- where $\boldsymbol{Q}$ is **orthogonal:**
  - $\boldsymbol{Q^{-1} = Q^T}$
  - Columns of $\boldsymbol{Q}$ are normalized eigenvectors
  - Columns are orthogonal.
  - (everything is real)

Exercise

- Examine the symmetric eigen decomposition, if any, for each of the following matrices:

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 \\ -2 & 3 \end{bmatrix} \quad \begin{bmatrix} 2 & 2 \\ 2 & 4 \end{bmatrix}$$

Time out!

- I came to this class to learn about text retrieval and mining, not have my linear algebra past dredged up again …
  - But if you want to dredge, Strang's *Applied Mathematics* is a good place to start.
- What do these matrices have to do with text?
- Recall $m \times n$ term-document matrices …
- But everything so far needs square matrices – so …

Singular Value Decomposition

For an $m \times n$ matrix $\mathbf{A}$ of rank r there exists a factorization (Singular Value Decomposition = **SVD**) as follows:

$$A = U \Sigma V^T$$

$m \times m$ | $m \times n$ | V is $n \times n$

The columns of $\boldsymbol{U}$ are orthogonal eigenvectors of $\boldsymbol{AA^T}$.

The columns of $\boldsymbol{V}$ are orthogonal eigenvectors of $\boldsymbol{A^TA}$.

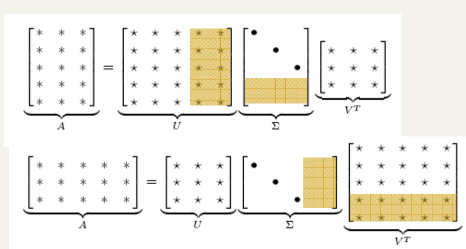
Eigenvalues $\lambda_1 \dots \lambda_r$ of $\boldsymbol{AA^T}$ are the eigenvalues of $\boldsymbol{A^TA}$.

$$\sigma_i = \sqrt{\lambda_i}$$

$$\Sigma = diag(\sigma_1 \dots \sigma_r)$$ *Singular values.*

Singular Value Decomposition

- Illustration of SVD dimensions and sparseness



SVD example

Let $A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$

Thus m=3, n=2. Its SVD is

$$\begin{bmatrix} 0 & 2/\sqrt{6} & 1/\sqrt{3} \\ 1/\sqrt{2} & -1/\sqrt{6} & 1/\sqrt{3} \\ 1/\sqrt{2} & 1/\sqrt{6} & -1/\sqrt{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{3} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

Typically, the singular values arranged in decreasing order.

Low-rank Approximation

- SVD can be used to compute optimal **low-rank approximations**.
- Approximation problem: Find $\boldsymbol{A_k}$ of rank $\boldsymbol{k}$ such that

$$A_k = \min_{X:rank(X)=k} \|A - X\|_F \longleftarrow \text{Frobenius norm}$$

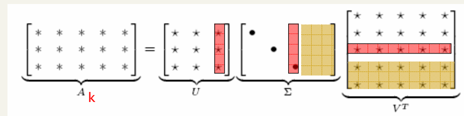
$$\|A\|_F \equiv \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}.$$

A_k and X are both $m \times n$ matrices.
Typically, want $k << r$.

Low-rank Approximation

- Solution via SVD

$$A_k = U \operatorname{diag}(\sigma_1, \ldots, \sigma_k, \underbrace{0, \ldots, 0}_{\text{set smallest r-k singular values to zero}})V^T$$



$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T \longleftarrow \text{column notation: sum of rank 1 matrices}$$

Approximation error

- How good (bad) is this approximation?
- It's the best possible, measured by the Frobenius norm of the error:

$$\min_{X:rank(X)=k} \|A - X\|_F = \|A - A_k\|_F = \sigma_{k+1}$$

where the σ_i are ordered such that $\sigma_i \geq \sigma_{i+1}$.
Suggests why Frobenius error drops as k increased.

Recall random projection

- Completely different method for low-rank approximation
- Was *data-oblivious*
  - SVD-based approximation is *data-dependent*
- Error for random projection depended only on start/finish dimensionality
  - For every distance
- Error for SVD-based approximation is for the Frobenius norm, not for individual distances

SVD Low-rank approximation

- Whereas the term-doc matrix A may have m=50000, n=10 million (and rank close to 50000)
- We can construct an approximation A_{100} with rank 100.
  - Of all rank 100 matrices, it would have the lowest Frobenius error.
- Great … but why would we??
- Answer: *Latent Semantic Indexing*

C. Eckart, G. Young, *The approximation of a matrix by another of lower rank.* Psychometrika, 1, 211-218, 1936.

Latent Semantic Analysis via SVD

What it is

- From term-doc matrix A, we compute the approximation A_k.
- There is a row for each term and a column for each doc in A_k
- Thus docs live in a space of $k \ll r$ dimensions
  - These dimensions are not the original axes
- But why?

Vector Space Model: Pros

- **Automatic** selection of index terms
- **Partial matching** of queries and documents *(dealing with the case where no document contains all search terms)*
- **Ranking** according to **similarity score** *(dealing with large result sets)*
- **Term weighting** schemes *(improves retrieval performance)*
- Various extensions
  - Document clustering
  - Relevance feedback (modifying query vector)
- Geometric foundation

Problems with Lexical Semantics

- Ambiguity and association in natural language
  - **Polysemy**: Words often have a **multitude of meanings** and different types of usage *(more urgent for very heterogeneous collections).*
  - The vector space model is unable to discriminate between different meanings of the same word.

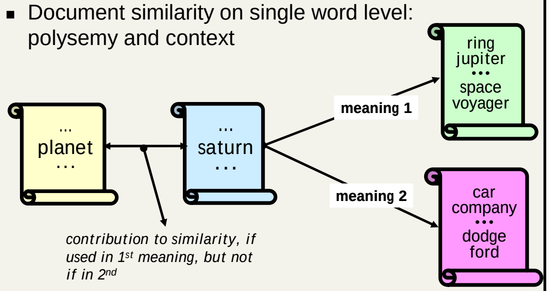
$$\mathrm{sim}_{\mathrm{true}}(d,q) < \cos(\angle(\vec{d},\vec{q}))$$

  - **Synonymy**: Different terms may have an **identical or a similar meaning** (weaker: words indicating the same topic).
  - No associations between words are made in the vector space representation.

$$\mathrm{sim}_{\mathrm{true}}(d,q) > \cos(\angle(\vec{d},\vec{q}))$$

Polysemy and Context

- Document similarity on single word level: polysemy and context

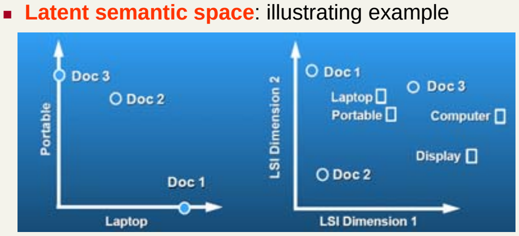


Latent Semantic Analysis

- Perform a **low-rank approximation** of **document-term matrix** (typical rank **100-300**)
- General idea
  - Map documents (*and* terms) to a **low-dimensional** representation.
  - Design a mapping such that the low-dimensional space reflects **semantic associations** (latent semantic space).
  - Compute document similarity based on the **inner product** in this **latent semantic space**
- Goals
  - Similar terms map to similar location in low dimensional space
  - Noise reduction by dimension reduction

Latent Semantic Analysis

- **Latent semantic space**: illustrating example



courtesy of Susan Dumais

Performing the maps

- Each row and column of *A* gets mapped into the *k*-dimensional LSI space, by the SVD.
- Claim – this is not only the mapping with the best (Frobenius error) approximation to *A*, but in fact *improves* retrieval.
- A query q is also mapped into this space, by

$$q_k = q^T U_k \Sigma_k^{-1}$$

  - Query NOT a sparse vector.

Empirical evidence

- Experiments on TREC 1/2/3 – Dumais
- Lanczos SVD code (available on netlib) due to Berry used in these expts
  - Running times of ~ one day on tens of thousands of docs
- Dimensions – various values 250-350 reported
  - (Under 200 reported unsatisfactory)
- Generally expect recall to improve – what about precision?

Empirical evidence

- Precision at or above median TREC precision
  - Top scorer on almost 20% of TREC topics
- Slightly better on average than straight vector spaces
- Effect of dimensionality:

Dimensions	Precision
250	0.367
300	0.371
346	0.374

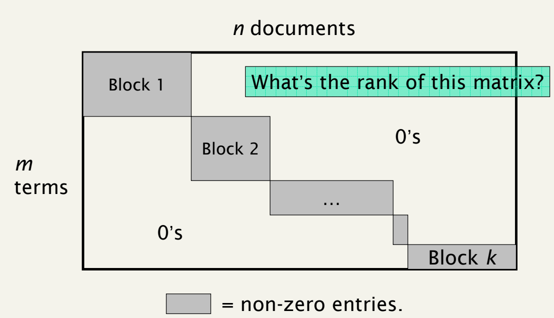
Failure modes

- Negated phrases
  - TREC topics sometimes negate certain query/terms phrases – automatic conversion of topics to
- Boolean queries
  - As usual, freetext/vector space syntax of LSI queries precludes (say) “Find any doc having to do with the following 5 companies”
- See Dumais for more.

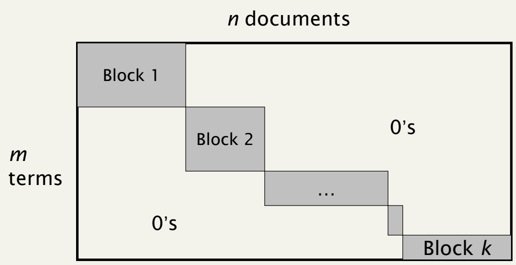
But why is this clustering?

- We’ve talked about docs, queries, retrieval and precision here.
- What does this have to do with clustering?
- Intuition: Dimension reduction through LSI brings together “related” axes in the vector space.

Intuition from block matrices

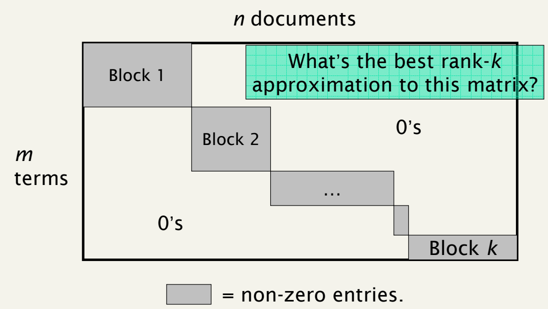


Intuition from block matrices

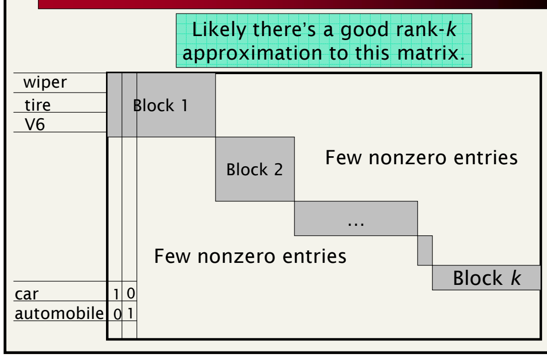


Vocabulary partitioned into *k* topics (clusters); each doc discusses only one topic.

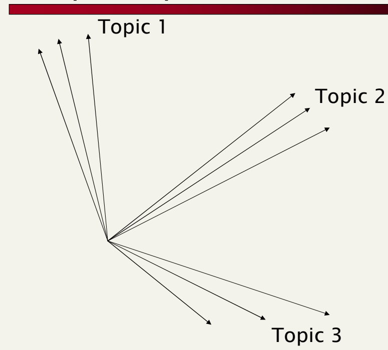
Intuition from block matrices



Intuition from block matrices



Simplistic picture



Some wild extrapolation

- The "dimensionality" of a corpus is the number of distinct topics represented in it.
- More mathematical wild extrapolation:
  - if *A* has a rank *k* approximation of low Frobenius error, then there are no more than *k* distinct topics in the corpus.

LSI has many other applications

- In many settings in pattern recognition and retrieval, we have a feature-object matrix.
  - For text, the terms are features and the docs are objects.
  - Could be opinions and users … more in 276B.
- This matrix may be redundant in dimensionality.
  - Can work with low-rank approximation.
  - If entries are missing (e.g., users' opinions), can recover if dimensionality is low.
- Powerful general analytical technique
  - Close, principled analog to clustering methods.

Resources

- http://www.cs.utk.edu/~berry/lsi++/
- http://lsi.argreenhouse.com/lsi/LSIpapers.html
- Dumais (1993) LSI meets TREC: A status report.
- Dumais (1994) Latent Semantic Indexing (LSI) and TREC-2.
- Dumais (1995) Using LSI for information filtering: TREC-3 experiments.
- M. Berry, S. Dumais and G. O'Brien. *Using linear algebra for intelligent information retrieval*. SIAM Review, 37(4):573--595, 1995.