Choosing Representation Size

A few practical issues

Representation "Accuracy"

- Recall Singular Value Decomposition
- $M_{m \times n} \approx \dot{M}_{m \times n} = U_{m \times p}\ \Sigma_{p \times p}\ V^T{}_{p \times n}$

- If $p = \min(m,n)$, then $M_{m \times n} = \dot{M}_{m \times n}$ but there is no compression

- Usually, we set $p <= \min(m,n)$, and compute only p columns of U and p rows of V^T

- SVD computes the "best" p vectors.

- The square of the matrix Σ shows how much each column of U (row of V^T) contributes to the approximation.

Representation "Accuracy"

- $M_{m\times n} \approx \dot{M}_{m\times n} = U_{m\times p}\, \Sigma_{p\times p}\, V^T{}_{p\times n}$

- Consider $p = 0$; assume that means $\dot{M} = 0$

- $(\|M_{m\times n} - \dot{M}_{m\times n}\|_F)^2 = (\|M_{m\times n}\|_F)^2$
  = sum of squares of elements of M
  call it SS

- This is the "worst" possible error.

Representation "Accuracy"

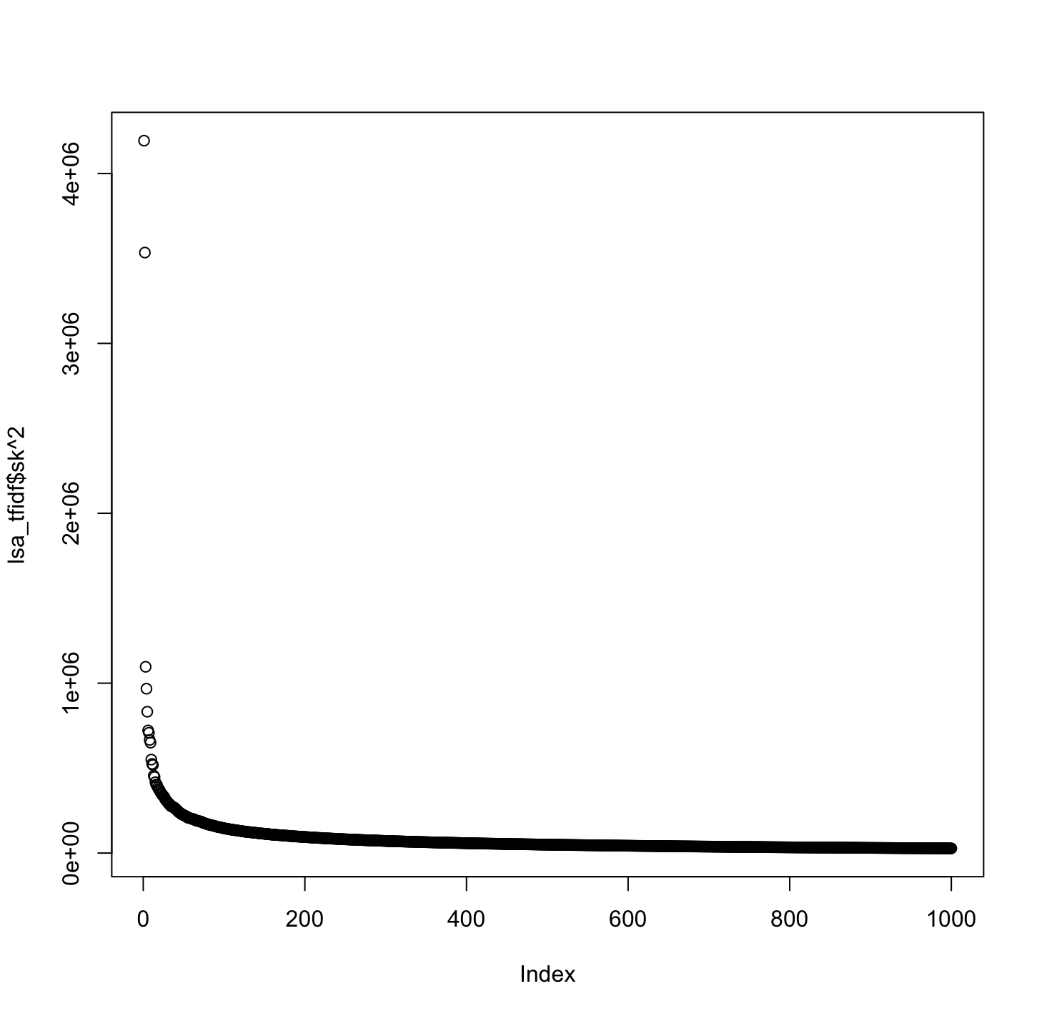
- $M_{m\times n} \approx \dot{M}_{m\times n} = U_{m\times p}\ \Sigma_{p\times p}\ V^T{}_{p\times n}$
- If $p = \min(m,n)$, then $M_{m\times n} = \dot{M}_{m\times n}$
- $(\|M_{m\times n} - \dot{M}_{m\times n}\|_F)^2 = (\ \|M_{m\times n} - M_{m\times n}\|_F)^2$

  $= 0$
- This is the "best" possible error.

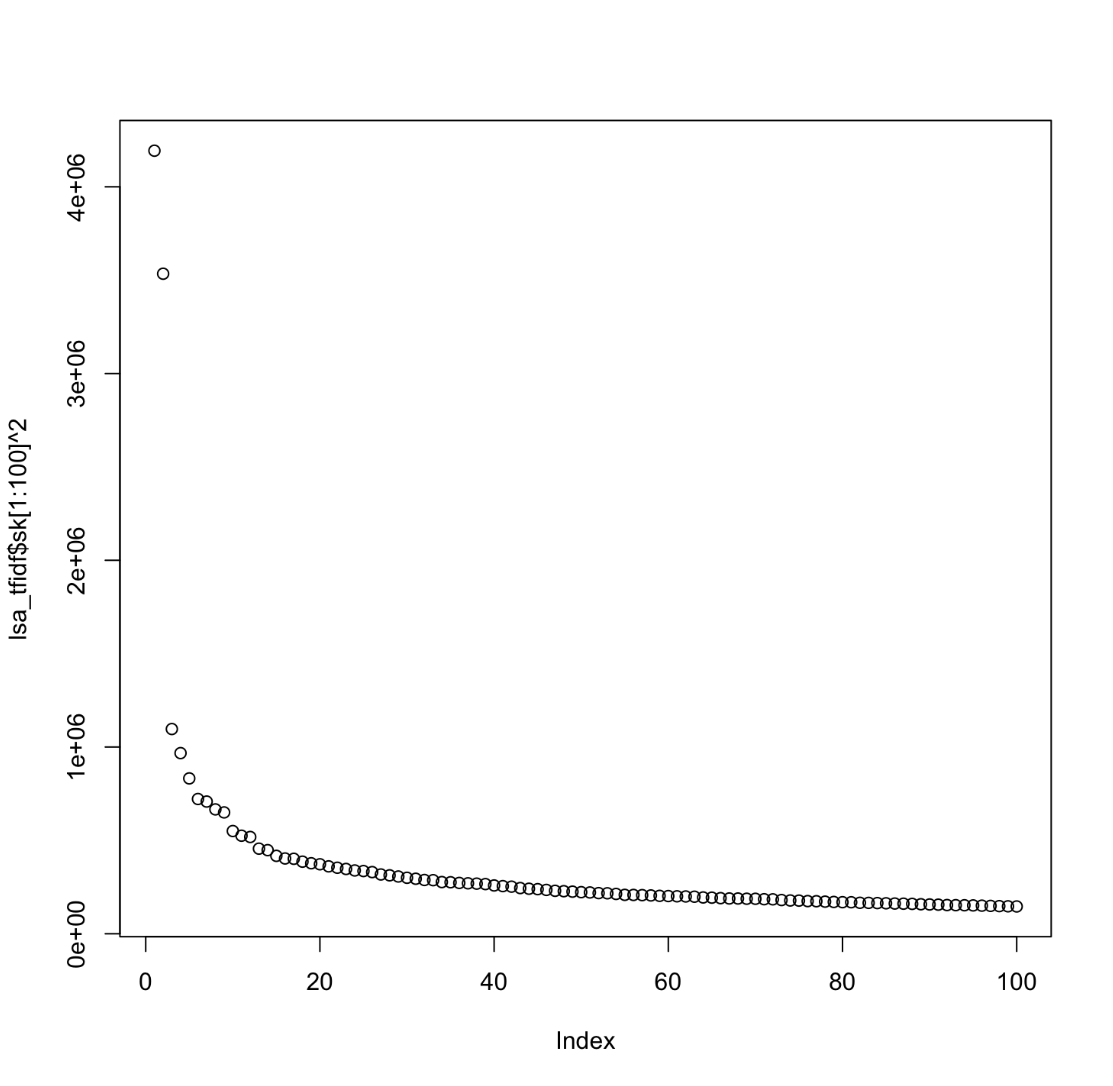
Representation "Accuracy"

- $\Sigma_{p \times p}$ is >= 0 on the diagonal, 0 everywhere else.
- Let's call its entries $\sigma_1, \sigma_2, \ldots, \sigma_p$
- σ^2_i tells us how much column *i* of U and row *i* of V^T improve the approximation (reduce the error)
- Also tell us the "importance" of topic *i*

Choosing *p*: "Elbow"

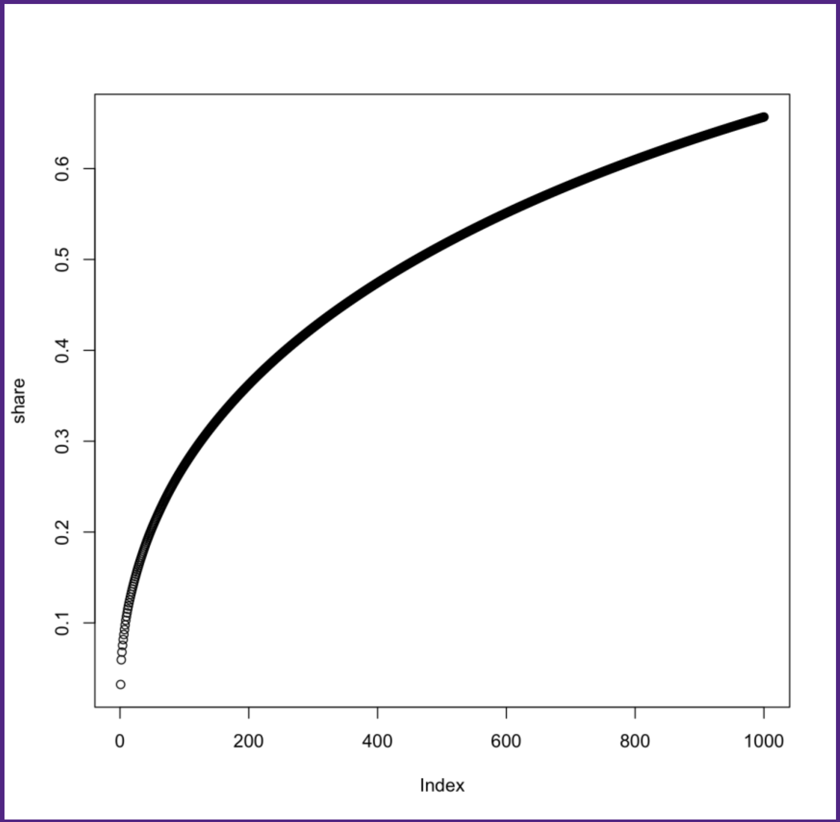


Choosing *p*: "Elbow"



Choosing *p*: “Share”

“Share” is cumulative sum of squared singular values, normalized by SS



p = 9 gives a share of 0.1; *p* = 48 gives a share of 0.2; *p* = 460 gives a share of 0.5

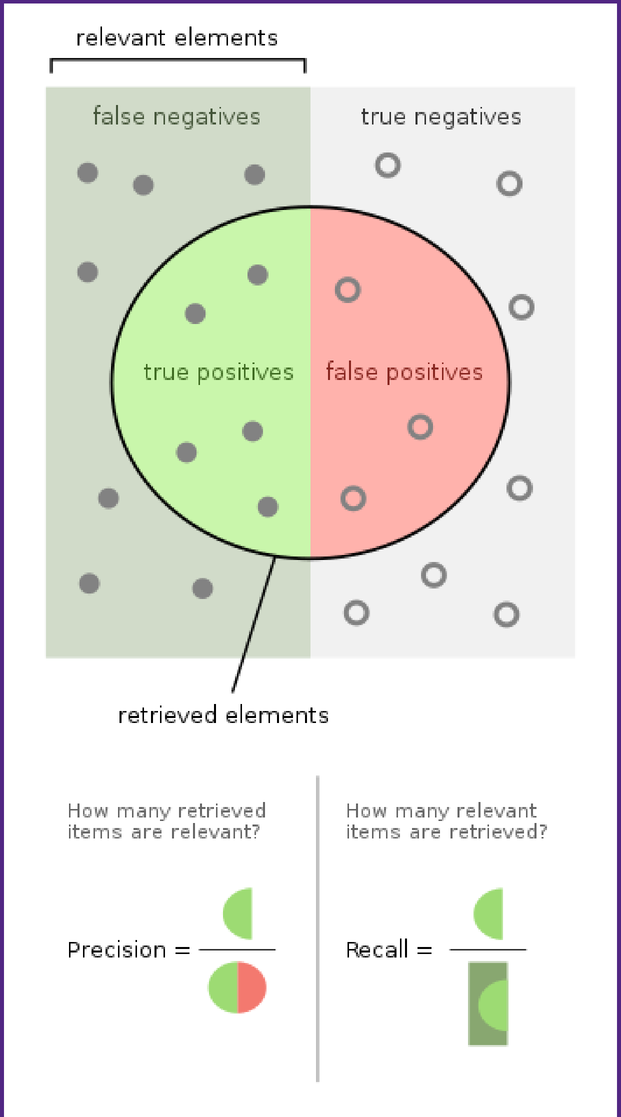
Choosing *p*: Application-driven

- "Why are we doing this again?"

- Examining topics manually to learn about the corpus
  - If you choose $p = 6$, the first 5 topics will be same as if you had chosen $p = 5$
  - As long as you are finding interesting topics, you can keep going

Choosing *p*: Application-driven

- If you are using the learned representations for retrieval, can evaluate different *p*
  - Precision = proportion of documents returned that are relevant
  - Recall = proportion of relevant documents in the corpus that are returned
  - F score = 2 * (precision*recall) / (precision + recall)
  - Between 0 and 1



Using the New Representations for Documents and Words

Rows of U and V as term and document representations

- $\dot{M}_{m \times n} = U_{m \times p} \, \Sigma_{p \times p} \, V^{T}{}_{p \times n}$

- Remember: *m* terms, *n* documents

- Each column of V^T corresponds to a document
- Each row of U corresponds to a term

- All of these vectors have dimension *p*.

- Each one can be used as a vector representation for a term or document

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

Columns of V^T are new vector representations for each document

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

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

Rows of U are new vector representations for each term

Re-arranging the SVD equation

- Recall: $V^TV = U^TU = I$ (identity)
- $\dot{M}_{m\times n} = U_{m\times p}\,\Sigma_{p\times p}\,V^T{}_{p\times n}$

- $\dot{M}_{m\times n}V_{n\times p} = U_{m\times p}\Sigma_{p\times p}V^T{}_{p\times n}V_{n\times p}$
- $\dot{M}_{m\times n}V_{n\times p} = U_{m\times p}\Sigma_{p\times p}I_{p\times p}$
- $\dot{M}_{m\times n}V_{n\times p} = U_{m\times p}\Sigma_{p\times p}$
- $\dot{M}_{m\times n}V_{n\times p}\Sigma^{-1}{}_{p\times p} = U_{m\times p}\Sigma_{p\times p}\Sigma^{-1}{}_{p\times p}$
- $\dot{M}_{m\times n}V_{n\times p}\Sigma^{-1}{}_{p\times p} = U_{m\times p}I_{p\times p}$

- $\dot{M}_{m\times n}(V_{n\times p}\Sigma^{-1}{}_{p\times p}) = U_{m\times p}$

Rows of U represent terms

- $\dot{M}_{m \times n} (V_{n \times p} \Sigma^{-1}{}_{p \times p}) = U_{m \times p}$

- Each element of row *i* of U is a weighted sum of a row of M

- Each element of row *i* of U is **summary** or a **feature** for **term** *i*

Re-arranging the SVD equation

- Recall: $V^TV = U^TU = I$ (identity)
- $\dot{M}_{m\times n} = U_{m\times p}\ \Sigma_{p\times p}\ V^T{}_{p\times n}$

- $U^T{}_{p\times m}\dot{M}_{m\times n} = U^T{}_{p\times m}U_{m\times p}\ \Sigma_{p\times p}\ V^T{}_{p\times n}$
- $U^T{}_{p\times m}\dot{M}_{m\times n} = I_{p\times p}\Sigma_{p\times p}\ V^T{}_{p\times n}$
- $U^T{}_{p\times m}\dot{M}_{m\times n} = \Sigma_{p\times p}\ V^T{}_{p\times n}$
- $\Sigma^{-1}{}_{p\times p}U^T{}_{p\times m}\dot{M}_{m\times n} = \Sigma^{-1}{}_{p\times p}\Sigma_{p\times p}\ V^T{}_{p\times n}$
- $\Sigma^{-1}{}_{p\times p}U^T{}_{p\times m}\dot{M}_{m\times n} = I_{p\times p}\ V^T{}_{p\times n}$

- $(\Sigma^{-1}{}_{p\times p}U^T{}_{p\times m})\dot{M}_{m\times n} = V^T{}_{p\times n}$

Rows of V represent documents

- $(\Sigma^{-1}_{p\times p}\, U^T_{p\times m})\dot{M}_{m\times n} = V^T_{p\times n}$

- Each element of column *j* of V^T is a weighted sum of a column of $\dot{M}$

- Each element of column *j* of V^T is **summary** or a **feature** for **document** *j*

SVD as dimensionality reduction

- In the term-document matrix,
  - Each term represented by a vector of length *n*
  - Each document represented by a vector of length *m*
  - These representations are not comparable.
- LSA/SVD gives us
  - Much more compact representations – length *p*
  - A **representation of all terms and documents *in the same space***
- Can use Cosine similarity, dot product, or other techniques like Euclidean distance, etc.
- Can compare document to document and document to term!

Retrieval

- Given a single-term query
  - Look up the corresponding row in U
  - Rank columns of V^T by their similarity to that row
  - Return documents in that order

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

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

Multi-word queries

- $\Sigma^{-1}{}_{p \times p}\, U^{T}{}_{p \times m}\, \dot{M}_{m \times n} = V^{T}{}_{p \times n}$
- This equation "shrinks" every document representation from length *m* to length *p*
- Given a new document $\mathbf{d}_{m \times 1}$, we get its representation like this:
- $\Sigma^{-1}{}_{p \times p}\, U^{T}{}_{p \times m} \mathbf{d}_{m \times 1} = \mathbf{v}_{p \times 1}$

Multi-word queries

- $\Sigma^{-1}{}_{p\times p}\, U^{T}{}_{p\times m}\mathbf{d}_{m\times 1} = \mathbf{v}_{p\times 1}$
- Given the new document's representation, compare it to all the representations in $V^{T}{}_{p\times n}$
- Can use cosine, for example
- Retrieve e.g. the top 10 most similar

Adding documents

- Representation depends on entire corpus
- To allow a new document to modify the representation (of all words and documents), must "re-compute" the SVD
- (There are algorithms for updating SVDs without a total re-do.)

Related Methods

Non-negative Matrix Factorization

- NMF for short
  - $\dot{M}_{m \times n} = W_{m \times p} H_{p \times n}$
  - Subject to $W_{m \times p} >= 0$, $H_{p \times n} >= 0$
- Similar (sometimes easier) interpretation because no negative weights.
- Cannot approximate the original matrix any better than SVD does. (Why?)
- Unlike SVD, not guaranteed to find global minimum.
- Topics not “ordered” by importance

Probabilistic
Latent Semantic Analysis

Summary

- LSA Compresses the Term Document Matrix

$$U_{m \times p} \, \Sigma_{p \times p} \, V^T_{p \times n}$$

- Columns of U represent word “clusters” (topics)
- Rows of V^T represent document “clusters”

- Rows of U represent words using p dimensions
- Columns of V^T represent documents using p dimensions

- Any technique that uses vector similarity or distance can use similarity or distance between rows of U and/or columns of V^T

- NMF gives similar output, $W_{m \times p} \, H_{p \times n}$ with restriction to positive values.
- PLSA gives probabilistic interpretation of topics

Retrieval Example