Taming text with SVD

Chennai Mathematical Institute

Presented by Aanchal, Aditi, Animesh, Aditya

July 2021

About

This presentation is about how SAS Text Miner use vector space model for representing text and how they use dimension reduction tools like SVD and PCA.

Introduction

- What do we mean by 'taming text'?

Introduction

- What do we mean by 'taming text'?
    - Giving mathematical form to a raw,unstructured text to get meaningful insights.

Introduction

- What do we mean by 'taming text'?
  - Giving mathematical form to a raw,unstructured text to get meaningful insights.
- But why do we need SVD here?

Introduction

- What do we mean by 'taming text'?
    - Giving mathematical form to a raw,unstructured text to get meaningful insights.
- But why do we need SVD here?
    - SAS Text Miner uses vector space model to represent text
    - Vector spaces are of high dimensions
    - That is why SVD is used for dimension reduction...

Vector Space model for Text

- To convert text to vector, SAS Text miner uses 'Bag of Words' approach.

Vector Space model for Text

- To convert text to vector, SAS Text miner uses 'Bag of Words' approach.

Lets elaborate this:

- Here we are calling a text document as document.
- Each distinct word of documents as item.
- We use set of items to form the 'Bag of Words'.

Vector Space model for Text Example

Suppose we have 3 documents. Number of times a term appears in each document are as follows:

Term	d1	d2	d3
"profile"	0	0	1
"frequency"	2	1	0
"mean"	1	0	1
"load"	0	0	1
"become"	1	1	0
"exit"	0	1	0

- This 'term document frequency matrix' is sparse as it has a lot of zeroes.

Dimension reduction

- In practical application, text mining uses lots of documents(in thousands).
- So, there are a lot of items in bag of words

Dimension reduction

- In practical application, text mining uses lots of documents(in thousands).
- So, there are a lot of items in bag of words
- So we get very high dimension to represent the documents
- As the size of collection of documents increases, the vector representation of a document becomes very sparse
- Because very few of the distinct terms in the collection are actually contained in any single document

Dimension reduction

- In practical application, text mining uses lots of documents(in thousands).
- So, there are a lot of items in bag of words
- So we get very high dimension to represent the documents
- As the size of collection of documents increases, the vector representation of a document becomes very sparse
- Because very few of the distinct terms in the collection are actually contained in any single document
- Handling this high dimensional space is much harder as matrices are of orders in thousands.
- So here we need to reduce dimension.

- Losing data : At the time of dimension reduction ,we should try to minimize loss of data. SAS Text Miner seeks to reduce dimension by:

Type

- Eliminating Terms
- Dimension reduction tools like SVD or PCA

Terms Elimination

SAS Text Miner provides 2 tools to eliminate the non informative terms and reduce dimension significantly. They are:

- Start lists: keep only a fixed, predetermined set of terms for the analysis. This is done by learning from previous analysis of similar data, and removing terms that were not useful in previous data.
- Roll up terms: restrict the number of variables to the specified number of highest weighted terms. This approach is dynamic, that is we assign weights to each term during the current analysis itself and take the highest weighted terms.

Some more term elimination techniques

Text Miner also includes a few other methods, which does not guarantee dimension reduction at a large scale like the previous 2 methods. They are:

- Stop lists: Removing a set of predetermined terms from analysis.
- Synonym lists: Remove terms that are similar by mapping one term to another.
- Stemming: Map multiple term to a root form.
- Remove singly occurring terms in the document.
- Remove numeric terms in the document

Problems with term co-occurrence, or why we need SVD or PCA

- Using the techniques mentioned in last 2 slides does not address problems like synonymy, polysemy, and term correlation.
- This leads to thousands of variables still in the data set which is impractical for any text mining solution.
- To address this problems, Text Miner provides SVD as an alternative.

SVD

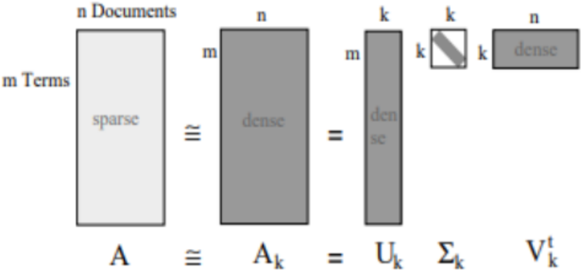
For recap

- If A be an $m \times n$ matrix, then singular value decomposition (SVD) of A is
$$A = U\Sigma V^t$$
- U is an m × r orthogonal matrix whose columns make up the left singular vectors of A
- Σ is an r × r dimensional diagonal matrix whose diagonal elements are termed singular values
- singular values are all greater than or equal to zero and, by convention, are ordered from largest to smallest
- V is an r × n orthogonal matrix whose columns form the right singular vectors of A

Approximations with the SVD

- Now suppose we want reduce dimension to some k < r
- This generates a rank k matrix, A_k, that is the best rank-k approximation to A in terms of least-squares best fit

$$A \approx A_k = U_k \Sigma_k V_k^t$$



Approximations with the SVD

$$A \approx A_k = U_k \Sigma_k V_k^t$$

- See here that the columns of U_k forms an orthonormal basis for the k-dimensional document space
- So, an m dimensional document d can be projected to k-dimensional space using this U_k as

$$\hat{d} = U_k^T d$$

- Note that ith row of U_k gives weight to the ith component of d to form $\hat{d}$
- let's see an example to make it clear

Approximations with the SVD

Here suppose our matrix A is the one which is shown before
Now after doing SVD on it for k=2
So our U_2 , Σ_2 and V_2 are

$$U_2 = \begin{bmatrix} -0.06 & 0.55 \\ -0.78 & -0.18 \\ 0.36 & 0.55 \\ 0.06 & 0.55 \\ 0.48 & -0.18 \\ 0.18 & -0.18 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 2.82 & 0 \\ 0 & 1.73 \end{bmatrix},$$

$$V_2^t = \begin{bmatrix} 0.84 & 0.51 & 0.17 \\ 0 & -0.32 & 0.95 \end{bmatrix}$$

Approximations with the SVD

Now, we obtain $\hat{A} = U_2^t \cdot A = \begin{bmatrix} 2.39 & 1.43 & 0.48 \\ 0 & -0.55 & 1.64 \end{bmatrix}$, which gives us the reduced document matrix.

Dividing each column vector by its norm, we get the final reduced term document matrix as:

$\hat{A} = \begin{bmatrix} 1 & 0.93 & 0.28 \\ 0 & -0.36 & 0.96 \end{bmatrix}$

We can now represent the original 6 dimensional matrix in 2 dimensions!

PCA

PCA is another approach for dimension reduction like SVD

- Suppose that A is our term document frequency matrix of $m \times n$
- Subtracting the mean of each column of A from each entry in that column to form B
- Covariance matrix $C = BB^T$
- Now $C = X\Lambda X^{-1}$ is eigenvalue decomposition
- From property of SVD we know that, each columns of X are left singular vector of B and root of each eigen vector are singular value

PCA

- eigen vectors X necessary for projecting the original document into the reduced space just as we done in SVD
- Now we sort these eigen vector and eigen values on their principal values,
  where ith principal value $= \frac{\lambda_i}{\sum_{j=1}^{r} \lambda_j}$
- For dimension reduction to k where k<r

$$\hat{d} = U_k^t \cdot d$$

where U_k is the eigenvector matrix with vectors arranged in decreasing order of principal values.

PCA

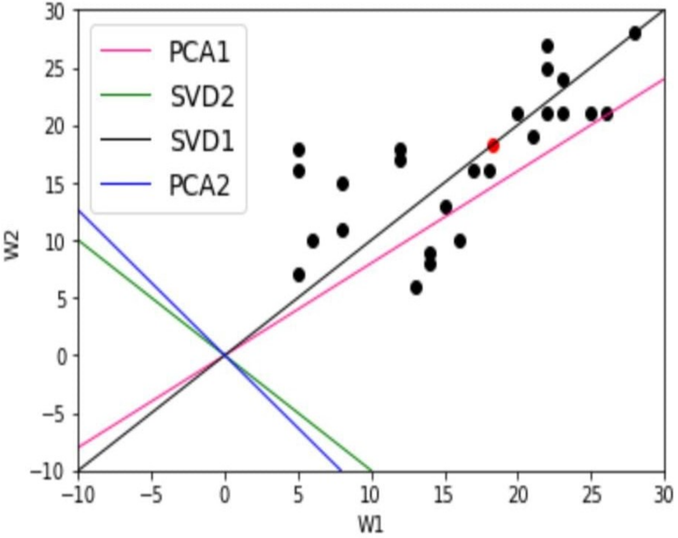
In PCA, we have the final matrix as:

$$\hat{A} = \begin{bmatrix} -0.96 & -0.63 & -1.59 \\ -0.79 & 0.91 & -0.12 \end{bmatrix}$$

Dividing each vector by its norm, we get the final reduced matrix as:

$$\hat{A} = \begin{bmatrix} -0.77 & -0.57 & -1 \\ -0.64 & 0.82 & -0.08 \end{bmatrix}$$

Plots of SVD and PCA



Pros and Cons of SVD

Advantage:

- In real data set, to use a data mining model, data reduction is an essential part. SVD comes handy in this situation in order to derive significantly fewer variables.

Pros and Cons of SVD

Advantage:

- In real data set, to use a data mining model, data reduction is an essential part. SVD comes handy in this situation in order to derive significantly fewer variables.

Drawbacks:

- Requires large amount of resource.
- Blurring effect on distinguishable documents due to linear combinations of input terms. So SVD fails when the discrimination is desired on finer level.
- Failed to analyze documents without rich interaction between terms and documents.

Criteria for choosing number of Dimensions

- Why the choice of number of dimension is essential?

Criteria for choosing number of Dimensions

- Why the choice of number of dimension is essential?
- How to choose the number of dimension?

Criteria for choosing number of Dimensions

- Why the choice of number of dimension is essential?
- How to choose the number of dimension?
  - For SVD, stop when the total variance of the data reached at a certain percentage by using sum of the first k v_i's obtained by the following formula.

$$v_i = \frac{\sigma_i^2}{\sum_{j=1}^{r} \sigma_j^2}.$$

Criteria for choosing number of Dimensions

- Why the choice of number of dimension is essential?
- How to choose the number of dimension?
  - For SVD, stop when the total variance of the data reached at a certain percentage by using sum of the first k v_i's obtained by the following formula.

$$v_i = \frac{\sigma_i^2}{\sum_{j=1}^{r} \sigma_j^2}.$$

  - For PCA, the equivalent formula is,

$$v_i = \frac{\lambda_i}{\sum_{j=1}^{r} \lambda_j}$$

Criteria for choosing number of Dimensions

- Why the choice of number of dimension is essential?
- How to choose the number of dimension?
  - For SVD, stop when the total variance of the data reached at a certain percentage by using sum of the first k v_i's obtained by the following formula.

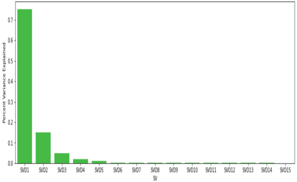
$$v_i = \frac{\sigma_i^2}{\sum_{j=1}^{r} \sigma_j^2}.$$

  - For PCA, the equivalent formula is,

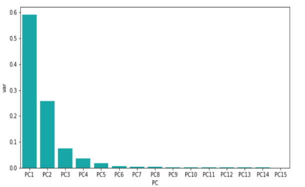
$$v_i = \frac{\lambda_i}{\sum_{j=1}^{r} \lambda_j}$$

Criteria for choosing number of Dimensions

- For SVD, the scree-plot is as follows.

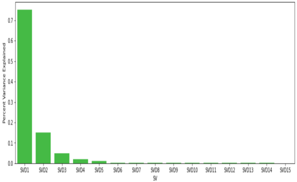


- For PCA, the scree-plot is as follows.

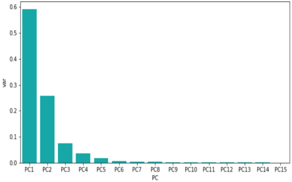


Criteria for choosing number of Dimensions

- For SVD, the scree-plot is as follows.



- For PCA, the scree-plot is as follows.



- Does these methods always work?

Significance of Dimensions

- What do these dimensions of the term-frequency matrix means?

Significance of Dimensions

- What do these dimensions of the term-frequency matrix means?
    - Loosely saying, here dimensions are a representation of words in the document, and the singular values represents the weight associated with a word, where components are linear combinations of original terms.

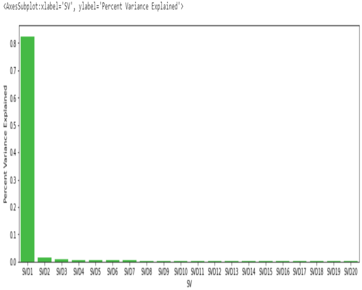
Significance of Dimensions

- What do these dimensions of the term-frequency matrix means?
    - Loosely saying, here dimensions are a representation of terms in the document, and the singular values represents the weight associated with a component, where components are linear combinations of original terms.
- Why TM only uses the SVD for dimension reduction?

Significance of Dimensions

- What do these dimensions of the term-frequency matrix means?
  - Loosely saying, here dimensions are a representation of words in the document, and the singular values represents the weight associated with a word, where components are linear combinations of original terms.
- Why TM only uses the SVD for dimension reduction?
  - PCA maps tens of thousands of variables to merely hundred variables, making it hard to interpret, whereas in SVD, after all the computation is complete we analyze the frequency of a particular term.

Observation



```
<AxesSubplot:xlabel='SV', ylabel='Percent Variance Explained'>
```

- If the data is too imbalanced then the first few singular/eigen values will dominate the other singular/eigen values, but it may happen the other eigen/singular values are significant.

Conclusions

- SAS Text Miner uses the vector space model for representing text.

Conclusions

- SAS Text Miner uses the vector space model for representing text.
- There are two possible way to reduce dimension.
    - Either eliminate terms.
    - Or Use SVD to generate new variables.

Conclusions

- SAS Text Miner uses the vector space model for representing text.
- There are two possible way to reduce dimension.
  - Either eliminate terms.
  - Or Use SVD to generate new variables.
- Formally SVD and PCA is computationally identical.
  - The only difference is PCA chooses orthogonal dimensions that optimally account for the variance in the data and SVD simply selects orthogonal dimensions that reduce the sum of the perpendicular distances from the original observations to the new axis.

Thank you! :-)