# MSc Data Mining and Machine Learning (2019)

# Lab 1 – Text Retrieval

|  |  |  |
| --- | --- | --- |
| Instructor: |  | Dr Peter Jancovic |
| Written by: |  | James J Nkhata |
|  |  |  |
| Experiment Date: |  | 04/02/2019 |
| Report Delivery Date: |  | 18/02/2019 |

## PART 2: LATENT SEMANTIC ANALYSIS

### Purpose

The aim of this part of the lab was to create a Word-document matrix from the folder of 112 “beng” documents and then have Latent Semantic Analysis (LSA) applied. The matrix created was from the “beng” folder documents that had stopping and stemming applied to them.

### Procedure

* + 1. Compiled the provided C file doc2vec to an executable .EXE using Microsoft Visual Studio Command prompt ‘cl doc2vec.c <ENTER>’ to generate doc2vec.exe
    2. Converted the StemFileList to a Matrix W in MATLAB:
       1. Using the command ‘doc2vec stemList.txt > WDM <ENTER>’, created document vector for the stemmed documents to a Matrix WDM
       2. In MATLAB using the command ‘>>W=load(‘WDM’)’, read the data in WDM into the MATLAB matrix W
    3. Applied Singular Value Decomposition (SVD) to matrix W:
       1. Using MATLAB command ‘>> [U, S, V] = svd(W)’, decomposed the matrix W into matrices U, S and VT

* + 1. Obtained the first three columns from matrix VT:
       1. Used command ‘>>sv1=V(:,1)’, to store matrix column 1 entries into column vector sv1
       2. Used command ‘>>sv2=V(:,2)’, to store matrix column 2 entries into column vector sv2
       3. Used command ‘>>sv3=V(:,3)’, to store matrix column 3 entries into column vector sv3
    2. Found the 1st, 2nd and 3rd biggest values in sv1 and their positions (index) in the vector to find their corresponding words:
       1. Used MATLAB command ‘>> m=min(sv1)’ to store the value of the biggest number in m (min was used in the command because the entries in sv1 were negative due to the nature of how MATLAB applies SVD)
       2. Used MATLAB command ‘[m, am] =min(sv1)’, to obtain the position ‘am’ (index) of the value of m in the vector sv1
       3. Exported sv1 to Microsoft Excel and reorder the column in smallest to largest order to find the 2nd and 3rd biggest numbers in sv1.
       4. Searched the sv1 vector (within MATLAB) for the indices obtained for the 2nd and 3rd biggest values in sv1
       5. Searched the index values found for the 1st, 2nd and 3rd biggest words of sv1 in ‘stemIndex’ file for the corresponding words
       6. Repeated step 2.2.5. for vectors sv2 and sv3 to find the three most significant words

### Experimental data

* + 1. Task 1 & 2: Creating the Word-document matrix and applying Singular Value Decomposition (SVD)

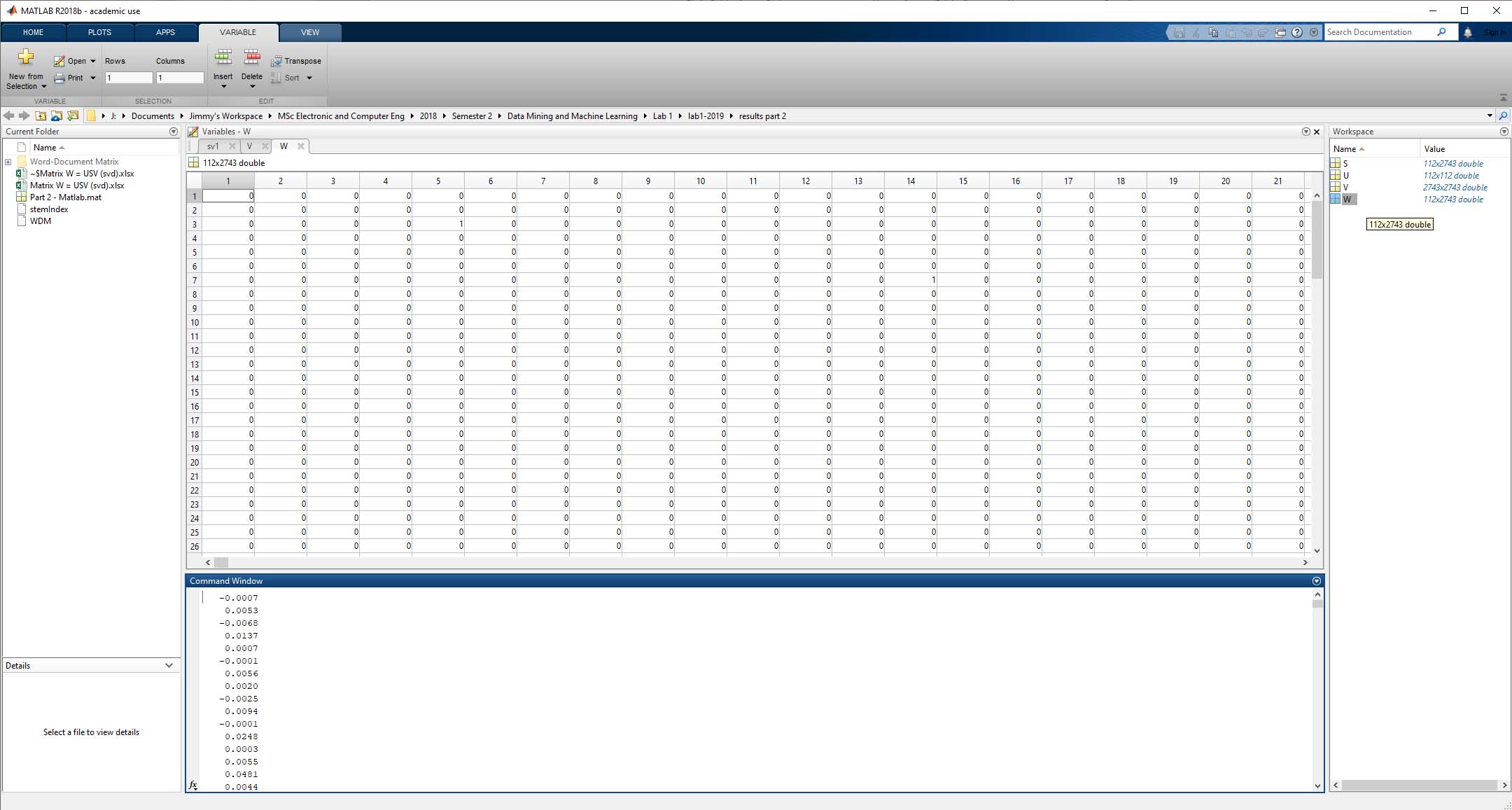


Figure 10: Applying Singular Value Decomposition (SVD) to the Word-document Matrix

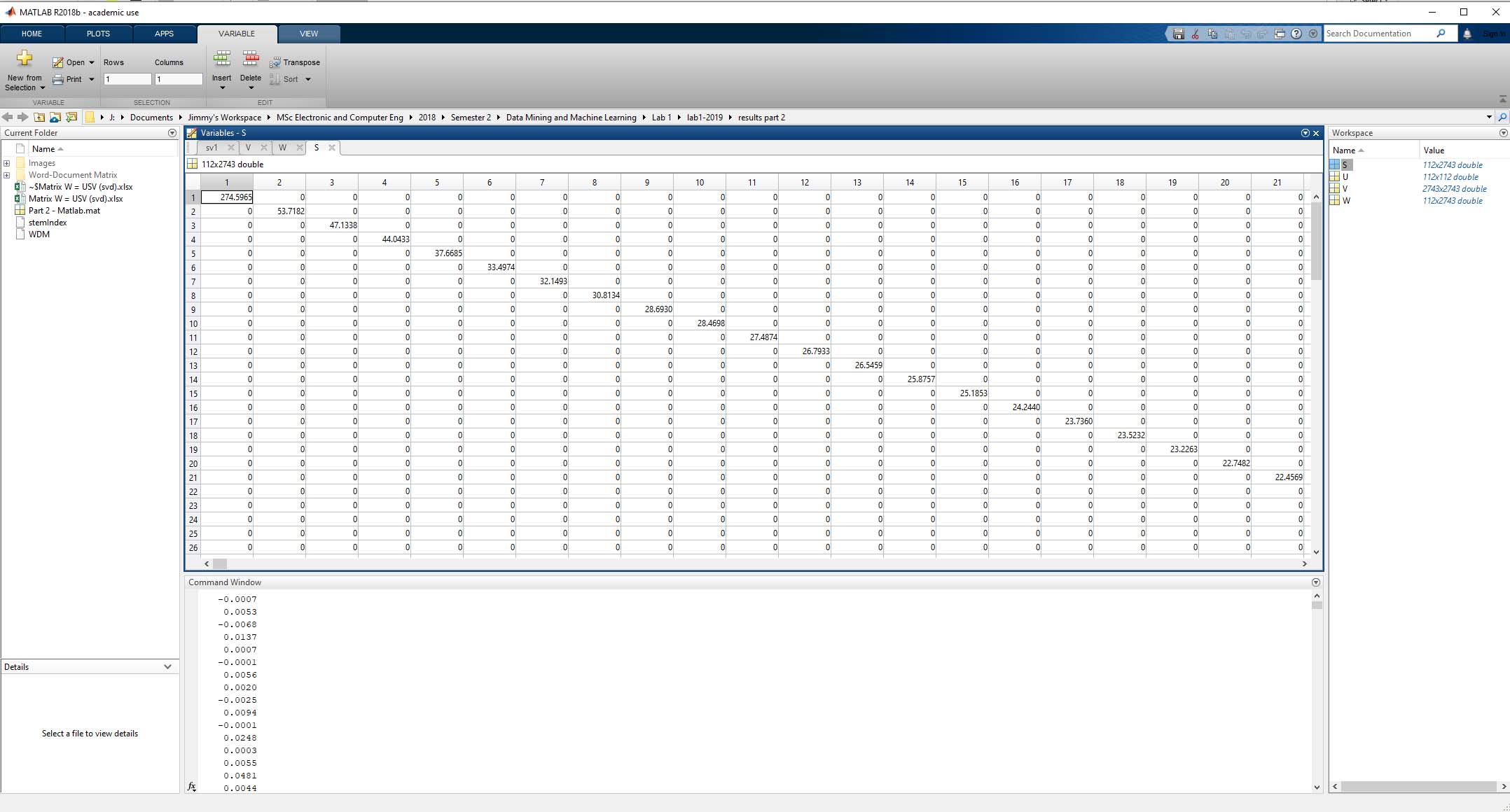


Figure 11: Singular Values Matrix S

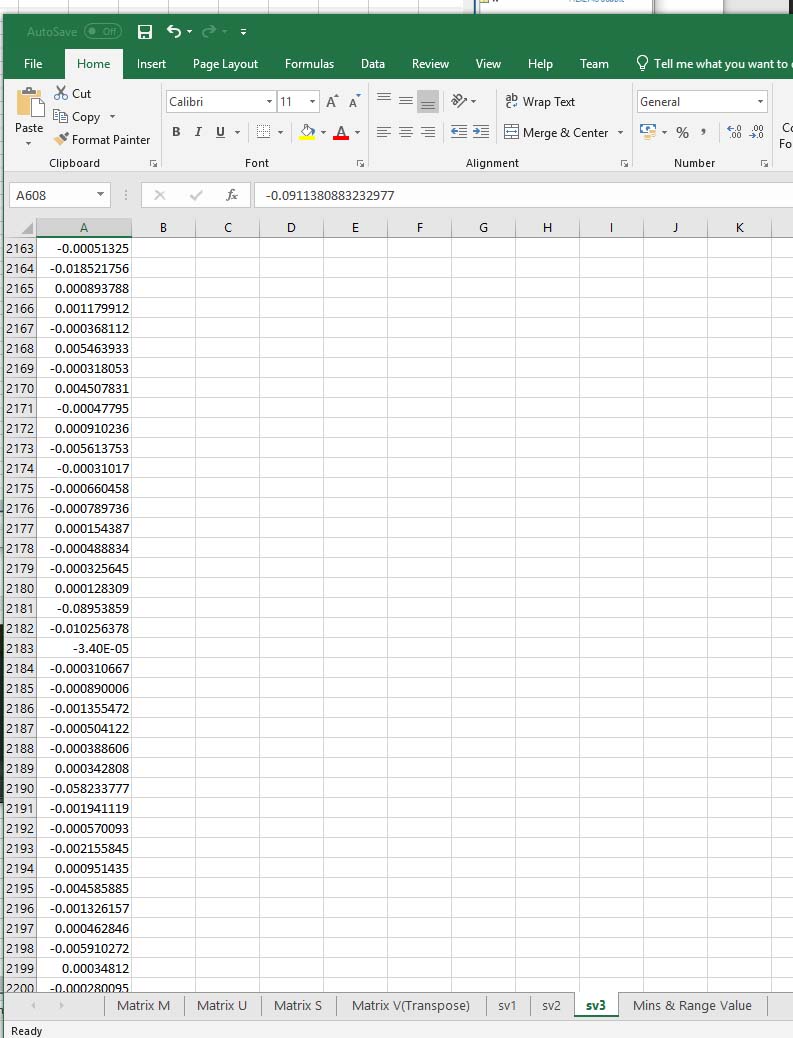


Figure 12: sv3 exported to Microsoft Excel (obtained from MATLAB Command sv3=V(:,3))

###### 

Figure 13: Biggest numbers in Vectors sv1, sv2 and sv3 (with their Indices Respectively)

### Questions

* + 1. Question 1:

Matrices U and V were as expected after applying the Singular value decomposition to Matrix W. Matrix U was of dimensions 112 x 112 which corresponded to the number of documents in the corpus and Matrix V was of dimensions 2743 x 2743 which corresponded to the vocabulary size.

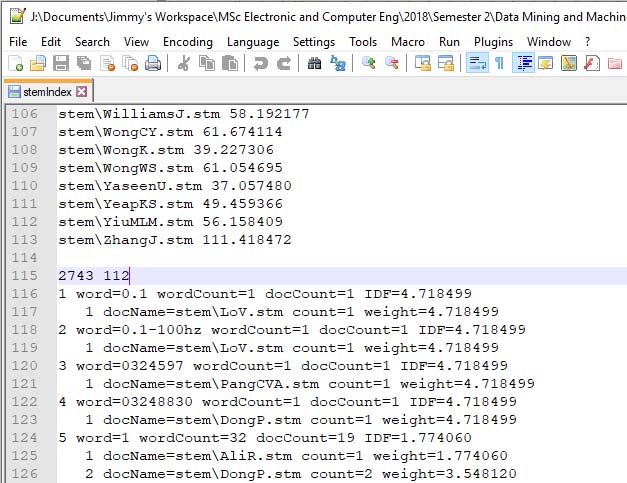


Figure 14: stemIndex (number of documents and vocabulary size)

The diagonal elements of S were ordered in descending order from largest to smallest

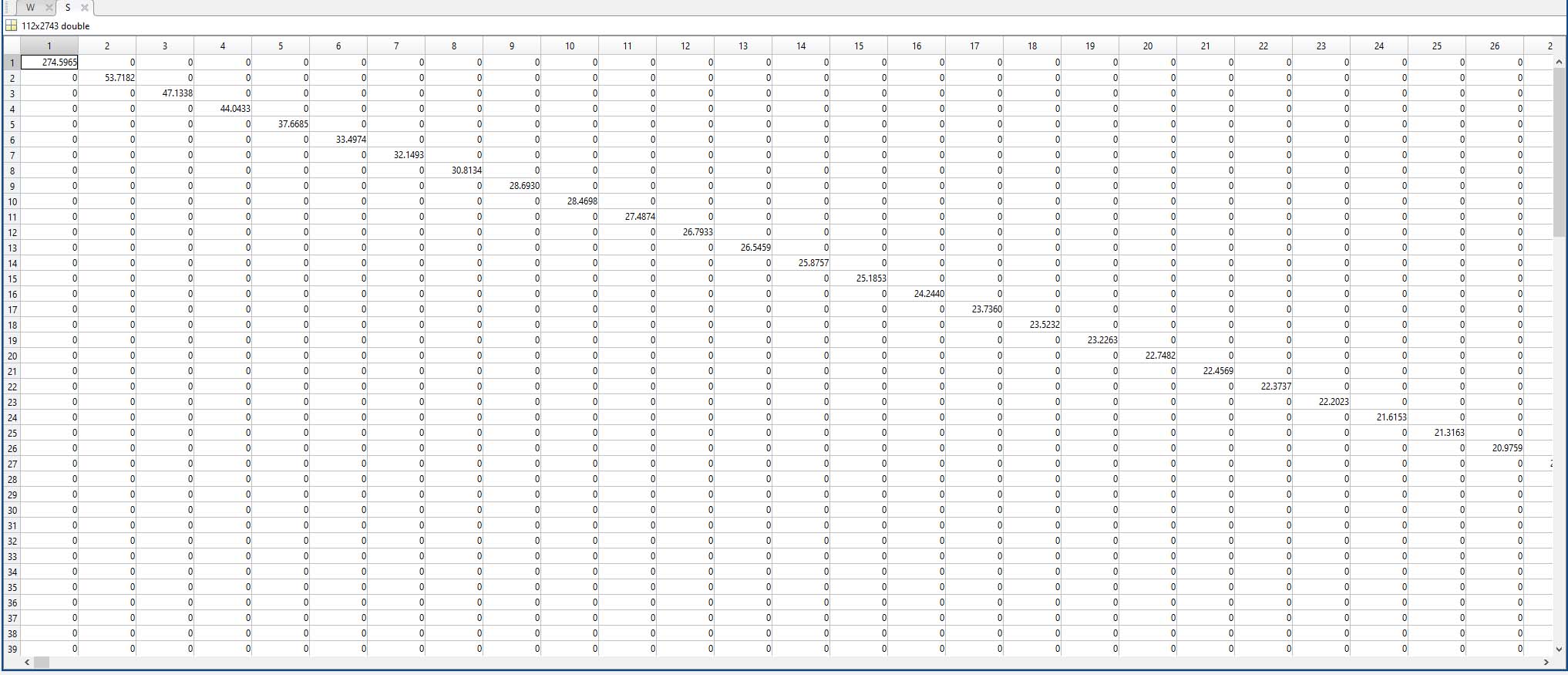


Figure 15:Diagonal values of Matrix S

* + 1. Question 2:

Values of the Diagonal Matrix S

Table 4: First three diagonal entries of Matrix S

|  |  |
| --- | --- |
| Matrix S | |
| Diagonal Element | Value |
| 1st | 274.5965 |
| 2nd | 53.7182 |
| 3rd | 47.1338 |

* + 1. Question 3:

The three most significant words for sv1 were “project” followed by “outcom” and then “student”

The three most significant words for sv2 were “data” followed by “reson” and then “coupl”

The three most significant words for sv3 were “speech” followed by “data” and then “should”

The results obtained from these words implied that the corpus was based on documents to do with a project undertaken by students on collecting speech related data.

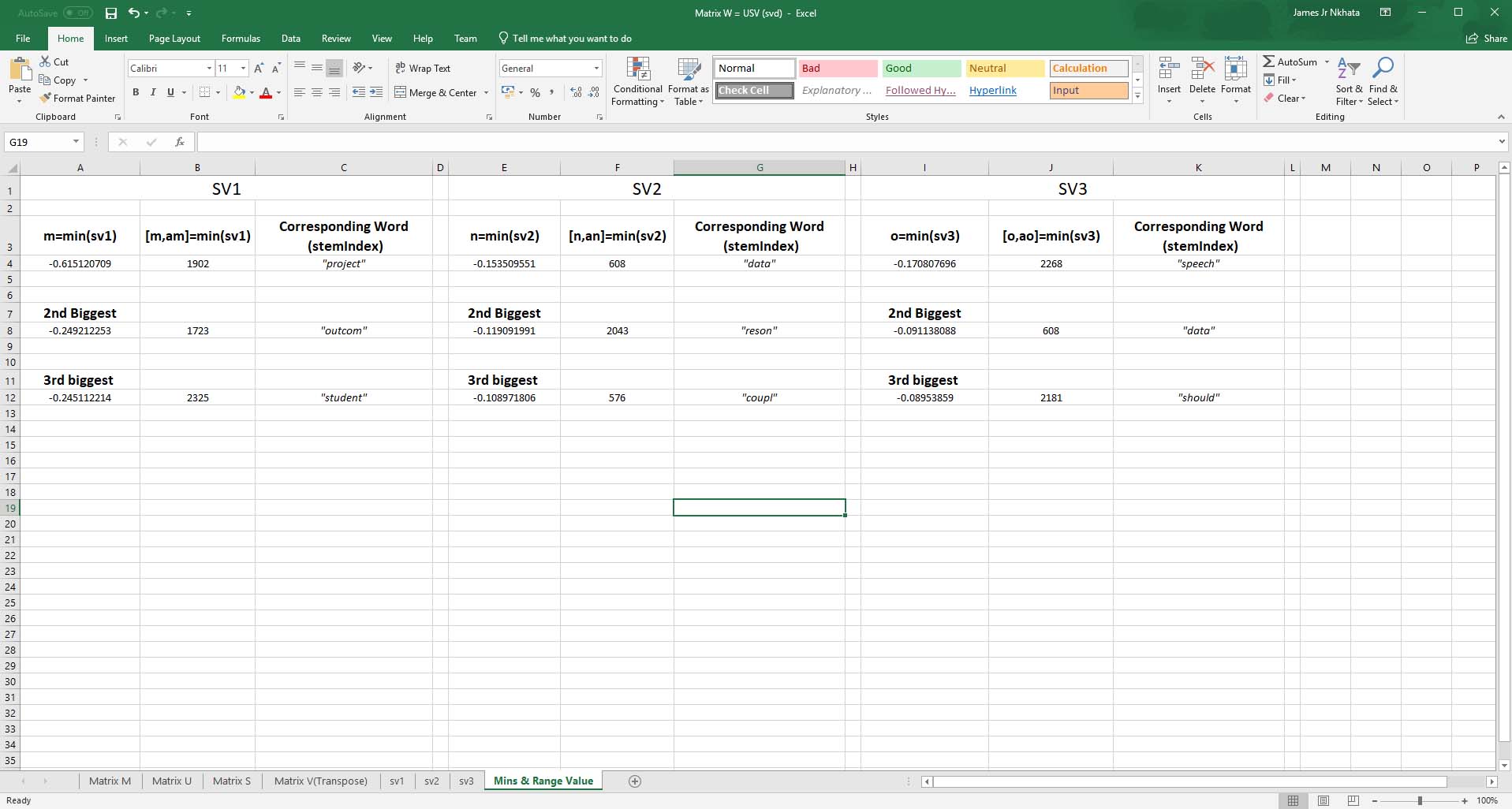


Figure 16: Results from Significant words from sv1, sv2 and sv3

### Conclusion

This lab report demonstrates how a Word-document matrix from the folder of 112 “beng” documents was created and Latent Semantic Analysis (LSA) was applied. The large corpus of the stemmed files (“stem” folder) were converted into a word-document matrix W (term frequency matrix) using doc2vec.exe which worked on the same concept as index.exe.

The results showed that the document-matrix was a matrix whose rows represented the terms and its columns represented the documents they were contained in. Therefore, matrix W represented the corresponding frequency of a term within the documents.

By using Singular Value Decomposition (SVD), the Matrix W was decomposed into matrices U, S and VT (V transpose). Matrix U held the correlation of documents to topic similarities, S held the strength of the topic (sorted in a decreasing diagonal format with non-diagonal elements being zero) and V held correlation of terms to topic similarities.

The columns of matrix V formed a coordinate system for the document vector space. The importance of a topic was correlated to its corresponding singular value in Matrix S.

In part 2 of the lab, the results obtained demonstrated how effective Latent Semantic Analysis can discover latent topics within a corpus through relationships between words.

### References

Belew, R.K. (2000) **Finding Out About: A cognitive Perspective on Search Engine Technology and the WWW.**1st ed. Cambridge: Cambridge University Press.

Jancovic, P (2019) **Data Mining and Machine Learning** [Online]. Available from: <https://canvas.bham.ac.uk/courses/34771> [Accessed 30 January 2019].