# Group Members:

XINGQUAN GUAN, UID: 804 759 254

DONG WANG, UID: 804 363 328

PENG-YU CHEN, UID: 704 732 316

ZHENGXIANG YI, UID: 304 761 570

# Acknowledgement:

All group members **have made equal contributions** to the project and same grading scale is expected for individual.

We accomplished this project using *Google Colab*. If anyone wants to test the submitted codes, please run it in *Google Colab* environment.

# Project Background:

This project is designed to help students get familiar with K-means clustering algorith. Through this process, some ticks to improve clustering performance, such as dimension reduction (using singular value decomposition and non-negative matrix factorization), scaling features, and logarithm transformation, are also involved and utilized. Additionally, several important concepts (including homogeneity, completeness, V-measure, adjusted rand index, adjusted mutual information score) relating to quantitative clustering performance evaluation are also introduced.

# Part 1:

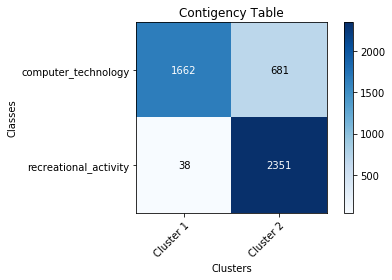
## Question 1: Report the dimensions of the TF-IDF matrix you get.

Firstly, the text data for *computer technology* and *recreational activity topics* from *20 news group dataset* is loaded. Then the data is vectorized using one-hot encoding scheme and the *term frequency-inverse document frequency* (TF-IDF) is computed based on the vectorized data.

The dimensions of TF-IDF matrix of training set are: (4732, 11389)

## Question 2: Report the contingency table of your clustering result.

After applying K-means clustering algorithm to the TF-IDF data, the contingency table for binary cluster is shown below:

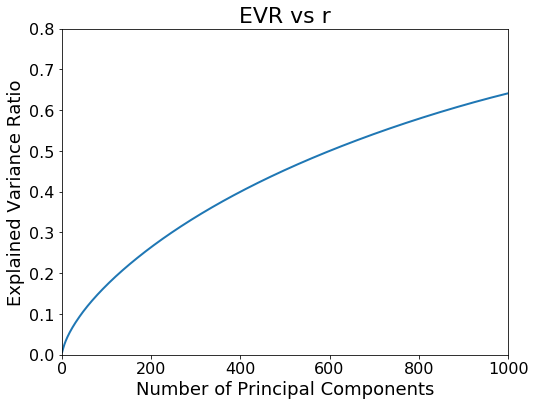


## Question 3: Report the 5 measures above for the K-means clustering results you get.

By calling the built-in Python package *sklearn.metrics*, the five measures are obtained and listed below:

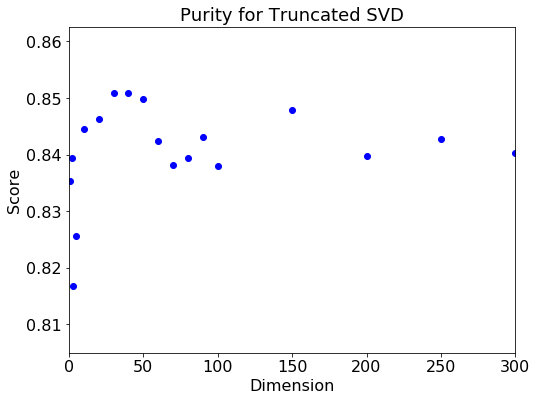
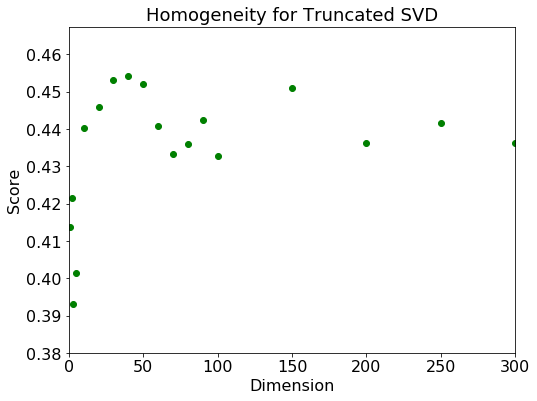
|  |  |
| --- | --- |
| Homogeneity | 0.452 |
| Completeness | 0.480 |
| V-measure | 0.466 |
| Adjusted rand index | 0.484 |
| Adjusted mutual information score | 0.452 |

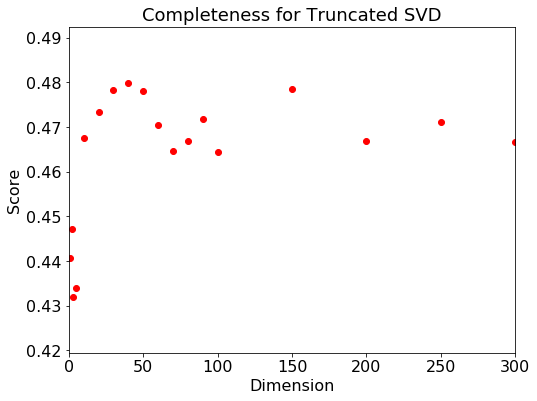
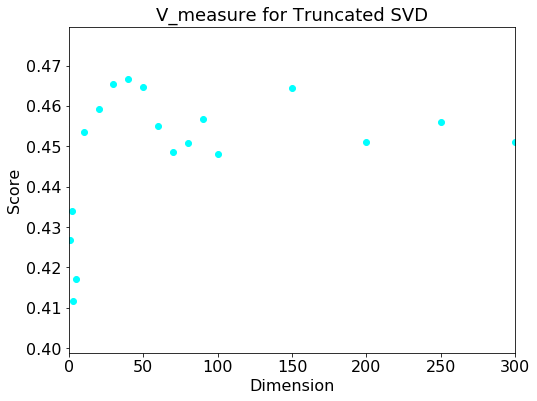
## Question 4: Report the plot of the percent of variance the top r principle components vs. r.

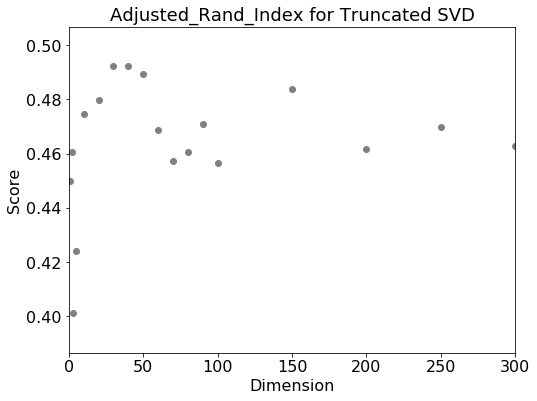


## Question 5: Report a good choice of r for SVD and NMF respectively.

1. SVD: The six measures (*purity* is added by our own purpose) with different number of components are plotted below. The figure title is attached at the top of each plot.

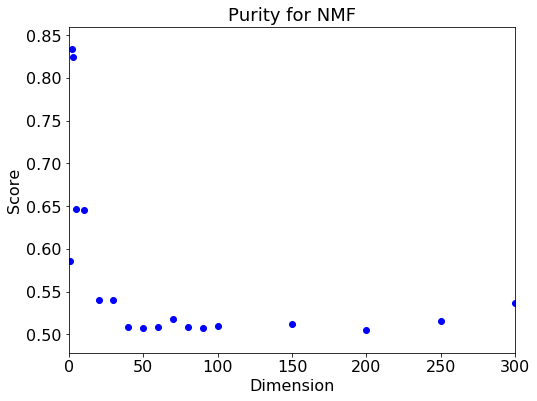
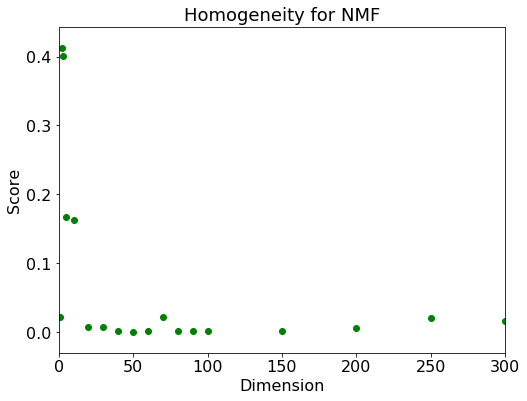
 

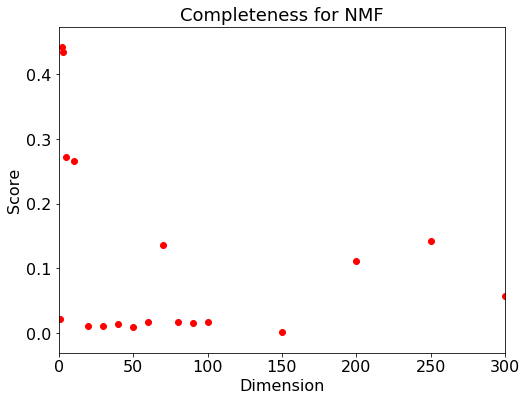
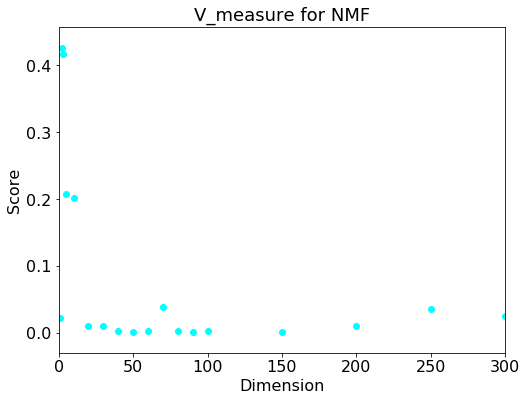
 

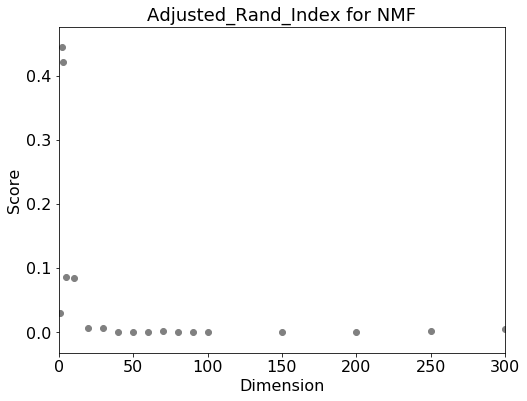
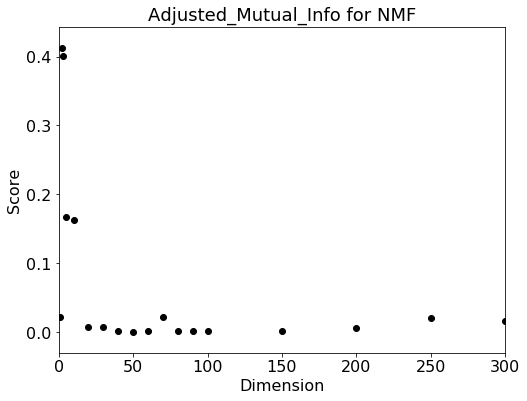
 

Based on six plots shown above, the best choice of *r* for SVD is 50.

1. NMF: The six measures (*purity* is added by our own purpose) with different number of components are plotted below. The figure title is attached at the top of each plot

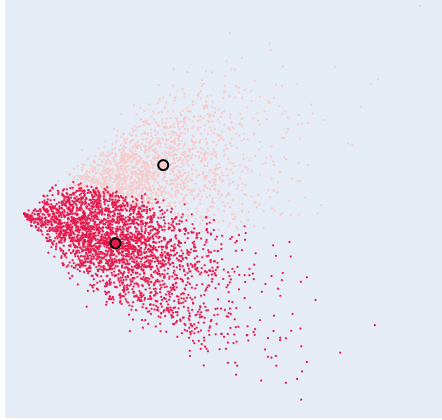
From the plots shown above, the best choice of *r* for NMF is 2.

## Question 6: How do you explain the non-monotonic behavior of the measures as r increases.

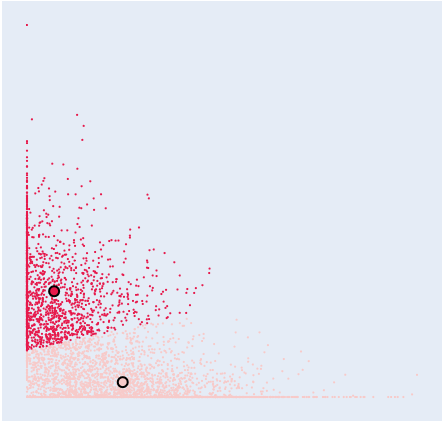
As more components are included, more information of data will be retained which will provide more corresponding effective information for clustering. However, when too many components are included and a high-dimensional space is thus created, the data will become very sparse and the distance measures will be very unreliable. The so-called “curse of dimensionality” happens. As a result, there is a point that strike a balance where features are enough for clustering while avoid the “curse of dimensionality”.

## Question 7: Visualize the clustering results for SVD and NMF with your choice of r.

1. SVD with best choice of *r*:

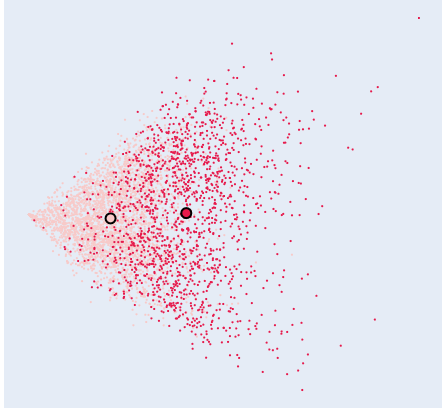


1. NMF with best choice of best *r*:

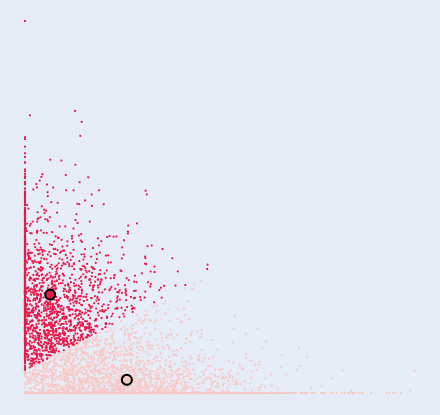


## Question 8: Visualize the transformed data.

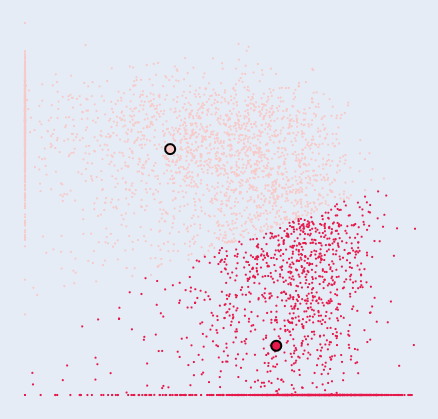
1. SVD with Scaling the features:



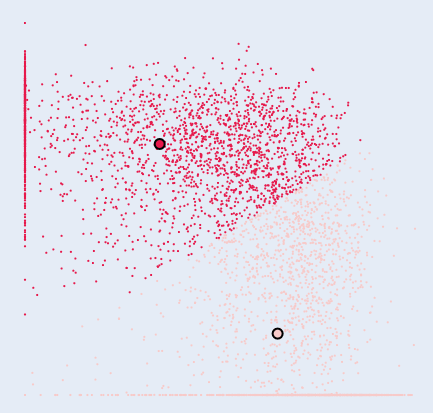
1. NMF with scaling features:



1. NMF with Logarithmic Transformation:



1. NMF with combination of scaling features and logarithmic transformation:



## Question 9: Can you justify why the “logarithm transformation” may improve the clustering results?

The plots in *Question 8* demonstrate that the two clusters are relatively close to each other before applying logarithm transformation. With logarithm transformation, the two clusters become “far” from each other and the boundary is obvious. The rationale behind this observation is that the logarithmic transformation can strengthen the similarity between data, especially in high-dimensional spaces where the distance measures can be decreased evidently.

## Question 10: Report the clustering measures (except for the contingency matrix) for the transformed data.

All required measured under different cases are listed in the table below.

1. SVD without scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.435 |
| Completeness | 0.466 |
| V-measure | 0.450 |
| Adjusted rand index | 0.459 |
| Adjusted mutual information score | 0.435 |

2. SVD with scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.019 |
| Completeness | 0.020 |
| V-measure | 0.019 |
| Adjusted rand index | 0.025 |
| Adjusted mutual information score | 0.019 |

3. NMF without scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.300 |
| Completeness | 0.308 |
| V-measure | 0.304 |
| Adjusted rand index | 0.362 |
| Adjusted mutual information score | 0.300 |

4. NMF with scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.459 |
| Completeness | 0.474 |
| V-measure | 0.467 |
| Adjusted rand index | 0.525 |
| Adjusted mutual information score | 0.459 |

5. NMF with logarithm transformation:

|  |  |
| --- | --- |
| Homogeneity | 0.454 |
| Completeness | 0.457 |
| V-measure | 0.455 |
| Adjusted rand index | 0.547 |
| Adjusted mutual information score | 0.453 |

6. NMF with combination of scaling and logarithm transformation:

|  |  |
| --- | --- |
| Homogeneity | 0.488 |
| Completeness | 0.488 |
| V-measure | 0.488 |
| Adjusted rand index | 0.596 |
| Adjusted mutual information score | 0.488 |

# Part 2:

# Dataset Introduction:

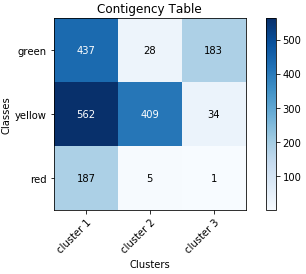
The dataset selected in this part is from our research project. It includes a number of descriptions about earthquake-induced damages on building structures and associated labels. Different descriptions correspond different labels. There are three labels in total and each one describes how severe the damage is. In this part, the K-means algorithm and associated tricks are applied on this dataset to see whether the K-means can successfully cluster those damage descriptions into three classes. Same analysis process in Part 1 is used herein. Please note that this dataset is extracted from engineering practice. Consequently, the clustering performance might not be as good as the ideal dataset (in Part 1).

## Question 1: Report the dimensions of the TF-IDF matrix you get.

The dimension for TF-IDF matrix is: (1846, 1995)

## Question 2: Report the contingency table of your clustering result.

After applying K-means clustering algorithm to the TF-IDF data, the contingency table for binary cluster is shown below:

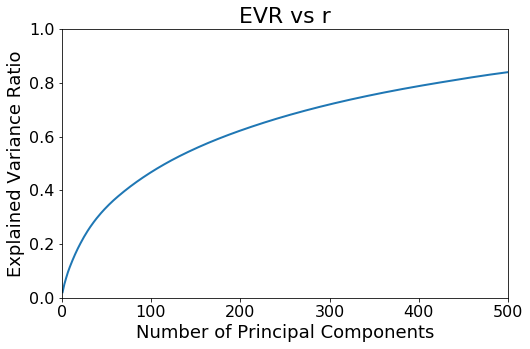


## Question 3: Report the 5 measures above for the K-means clustering results you get.

By calling the built-in Python package *sklearn.metrics*, the five measures are obtained and listed below:

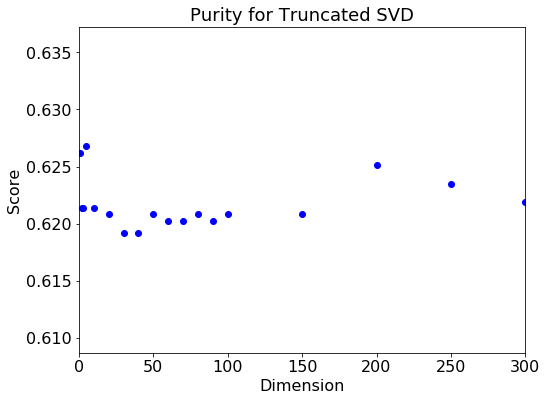
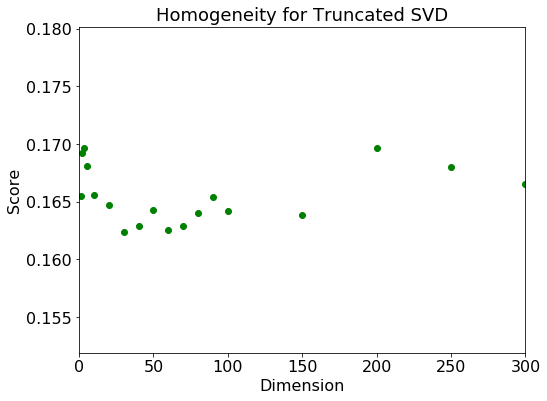
|  |  |
| --- | --- |
| Homogeneity | 0.169 |
| Completeness | 0.180 |
| V-measure | 0.174 |
| Adjusted rand index | 0.040 |
| Adjusted mutual information score | 0.168 |

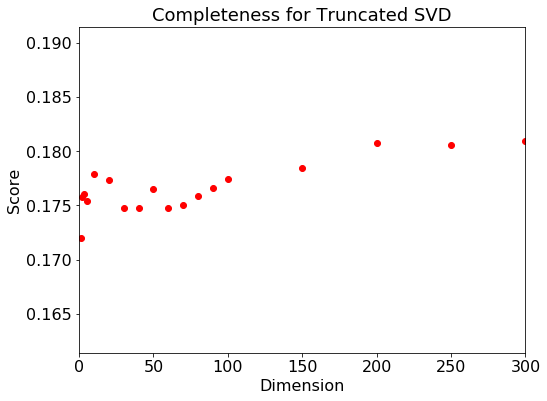
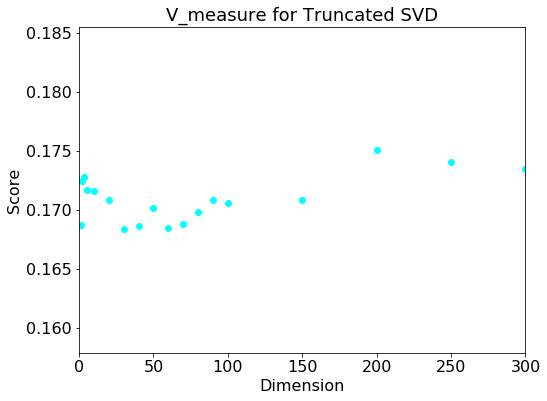
## Question 4: Report the plot of the percent of variance the top r principle components vs. r.

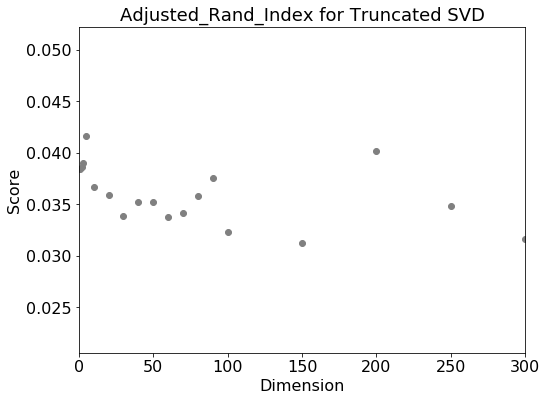
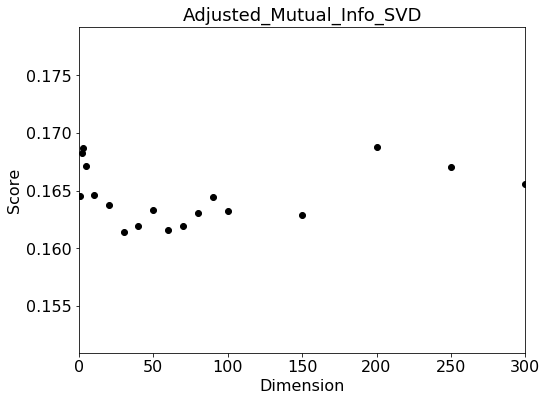


## Question 5: Report a good choice of r for SVD and NMF respectively.

1. SVD: The six measures (*purity* is added by our own purpose) with different number of components are plotted below. The figure title is attached at the top of each plot.

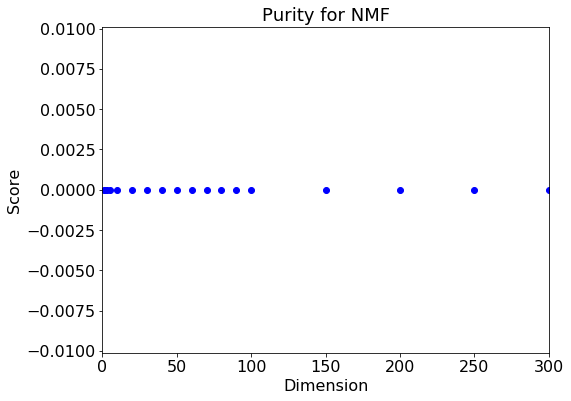
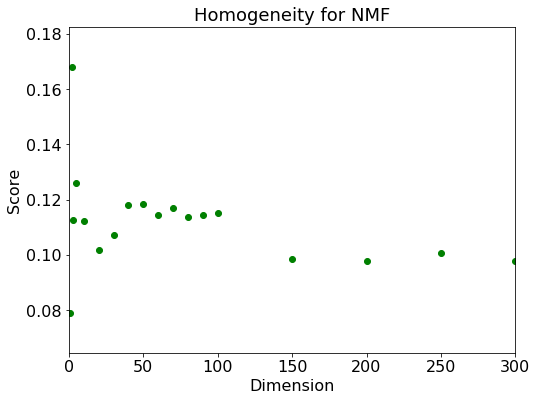
 

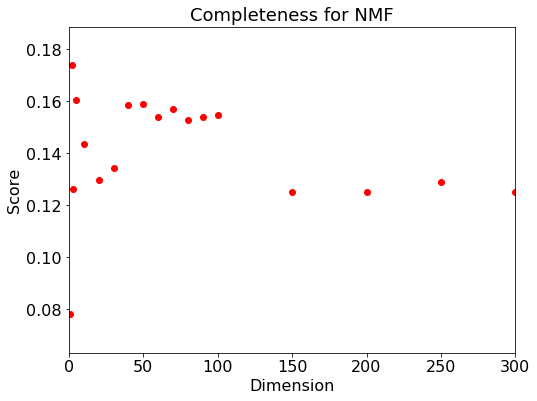
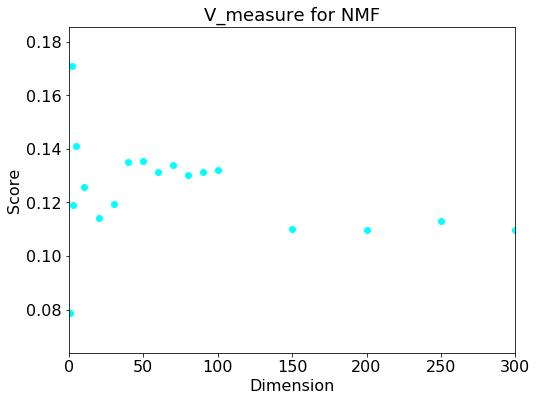
 

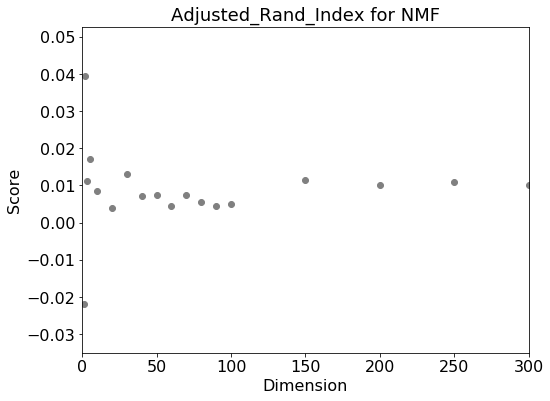
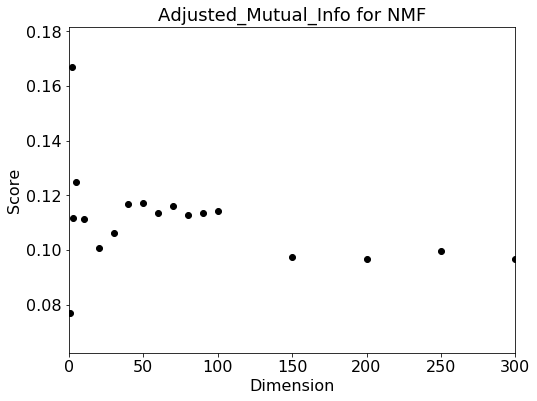
 

Based on the plots above, the best choice of *r* value for SVD is 5.

1. NMF: The six measures (*purity* is added by our own purpose) with different number of components are plotted below. The figure title is attached at the top of each plot

Based on the plots above, the best choice of *r* value for NMF is 2.

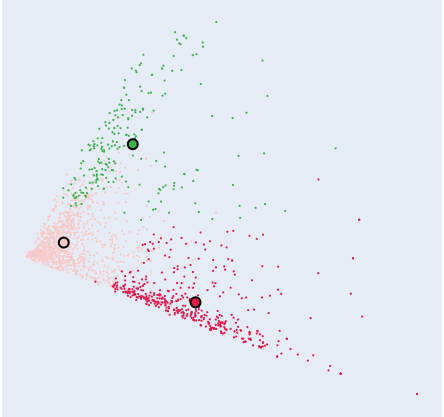
## Question 6: How do you explain the non-monotonic behavior of the measures as r increases.

When *r* value is relatively small, the number of features are too small to support the K-means algorithm for classifying the data points. However, when r is too large, the number of features is too great and the "curse of dimensionality" occurs. Between these two extremes, there is a balance point when the r is neither too large nor too small such that the number of features is enough to distinguish different clusters while the "curse of dimensionality" does not happen.

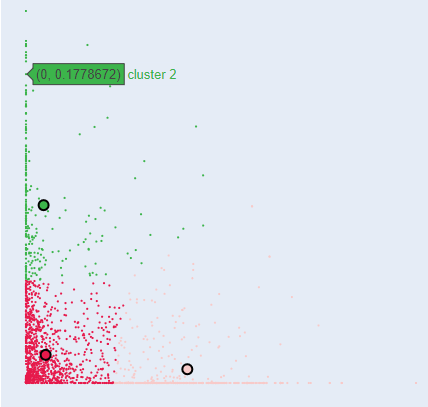
## Question 7: Visualize the clustering results for SVD and NMF with your choice of r.

Best r for SVD is 5 and for NMF is 2

1. SVD with best choice of *r*:

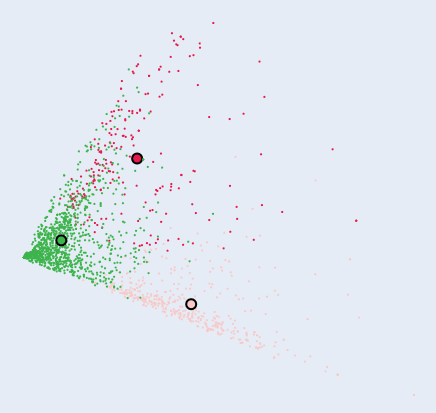


1. NMF with best choice of *r*:

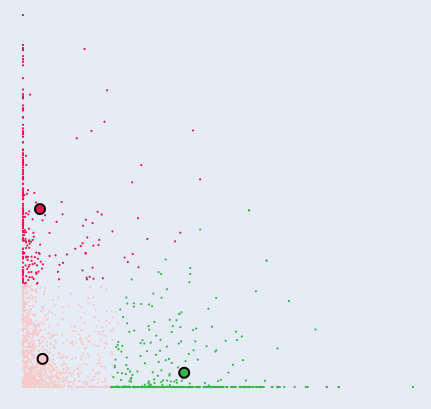


## Question 8: Visualize the transformed data.

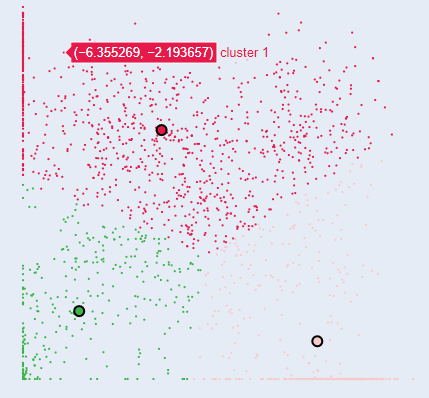
1. SVD after scaling the features:



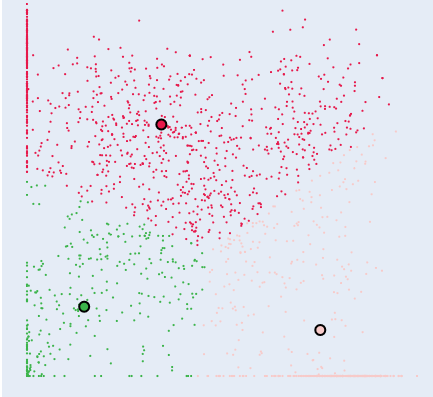
1. NMF after scaling the features:



1. NMF after logarithm transformation:



1. NMF after the combination of scaling features and logarithm transformation:



## Question 9: Can you justify why the “logarithm transformation” may improve the clustering results?

The plots in *Question 8* demonstrate that the two clusters are relatively close to each other before applying logarithm transformation. With logarithm transformation, the two clusters become “far” from each other and the boundary is obvious. The rationale behind this observation is that the logarithmic transformation can strengthen the similarity between data, especially in high-dimensional spaces where the distance measures can be decreased evidently.

## Question 10: Report the clustering measures (except for the contingency matrix) for the transformed data.

All required measured under different cases are listed in the table below.

1. SVD without scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.169 |
| Completeness | 0.179 |
| V-measure | 0.174 |
| Adjusted rand index | 0.037 |
| Adjusted mutual information score | 0.173 |

2. SVD with scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.137 |
| Completeness | 0.148 |
| V-measure | 0.142 |
| Adjusted rand index | 0.019 |
| Adjusted mutual information score | 0.141 |

3. NMF without scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.168 |
| Completeness | 0.174 |
| V-measure | 0.171 |
| Adjusted rand index | 0.040 |
| Adjusted mutual information score | 0.170 |

4. NMF with scaling the features:

|  |  |
| --- | --- |
| Homogeneity | 0.169 |
| Completeness | 0.175 |
| V-measure | 0.172 |
| Adjusted rand index | 0.041 |
| Adjusted mutual information score | 0.171 |

5. NMF with logarithm transformation:

|  |  |
| --- | --- |
| Homogeneity | 0.118 |
| Completeness | 0.109 |
| V-measure | 0.113 |
| Adjusted rand index | 0.088 |
| Adjusted mutual information score | 0.112 |

6. NMF with combination of scaling and logarithm transformation:

|  |  |
| --- | --- |
| Homogeneity | 0.123 |
| Completeness | 0.113 |
| V-measure | 0.118 |
| Adjusted rand index | 0.100 |
| Adjusted mutual information score | 0.117 |

# Part 3:

## Question 11: BONUS: can you suggest a methodology to make an appropriate choice of k and initial seeds of cluster centers?

1. Original Image



1. Color Clustering



If we talk about K-Means then the correct choice of K is often ambiguous, with interpretations

depending on the shape and scale of the distribution of points in a data set and the desired clustering resolution of the user. In addition, increasing K without penalty will always reduce the amount of error in the resulting clustering, to the extreme case of zero error if each data point is considered its own cluster (i.e., when K equals the number of data points, n). Intuitively then, the optimal choice of K will strike a balance between maximum compression of the data using a single cluster, and maximum accuracy by assigning each data point to its own cluster.

If an appropriate value of K is not apparent from prior knowledge of the properties of the data set, it must be chosen somehow. There are several categories of methods for making this decision and Elbow method is one such method.