Project 2: Clustering

Name: Jianxiong Wang, Yijun Wu, Yanzhao Wang, Yutong Sun

Date: 2019.2.03

Question 1

Report the dimensions of the TF-IDF matrix you get.

The dimension of TF-IDF matrix we get from ‘20 News group’ dataset is 7882 x 27768 (7882 documents with 27768 terms) using TfidfVectorizer object with min_df = 3.

Question 2

Report the contingency table of your clustering result.

Before we build the K-Means model and fit the model to the TF-IDF matrix, we first change the labels (categories) of the data from 0-7 to 0 and 1 since we are using k-means clustering with k = 2.

Then we use sklearn.cluster.KMeans object with n_clusters = true_k, random_state=0, max_iter=1000, n_init=30 to build the k-means model. After we fit the model to the data, we use contingency_matrix, homogeneity_score, completeness_score, v_measure_score, adjusted_rand_score, adjusted_mutual_info_score from sklearn.metrics.cluster to get the contingency table, and 5 measures (shown in Question 3) as following:

$$\begin{bmatrix} 4 & 3899 \\ 1718 & 2261 \end{bmatrix}$$

Question 3

Report the 5 measures above for the K-means clustering results you get.

Measure	Score
Homogeneity:	0.253596
Completeness:	0.334816
V-measure:	0.288600
Adjusted Rand-Index:	0.180762
Adjusted Mutual Information Score:	0.253528

Table 1. 5 Measures for the K-means Clustering with K = 2 and the TF-IDF Dataset

Question 4

Report the plot of the percent of variance the top r principal components can retain v.s. r, for r = 1 to 1000.

To get the percent of variance of the truncated SVD representation, we first reduce the dimensions of TF-IDF matrix from 27768 to 1000 using sklearn.decomposition.TruncatedSVD with n_components = 1000. Then we use the explained_variance_ratio_ attribute in TruncatedSVD object to get the variance each dimension in the reduced matrix can keep from the total variance of the original data. To calculate the percent of variance the top r principal components can maintain, we compute the prefix sum array of the explained_variance_ratio_ array, and the percent of variance of the top r components is the value at index r - 1 in the prefix sum array. Then, we plot the percent of variance of variance corresponding to r as Figure 1 below.

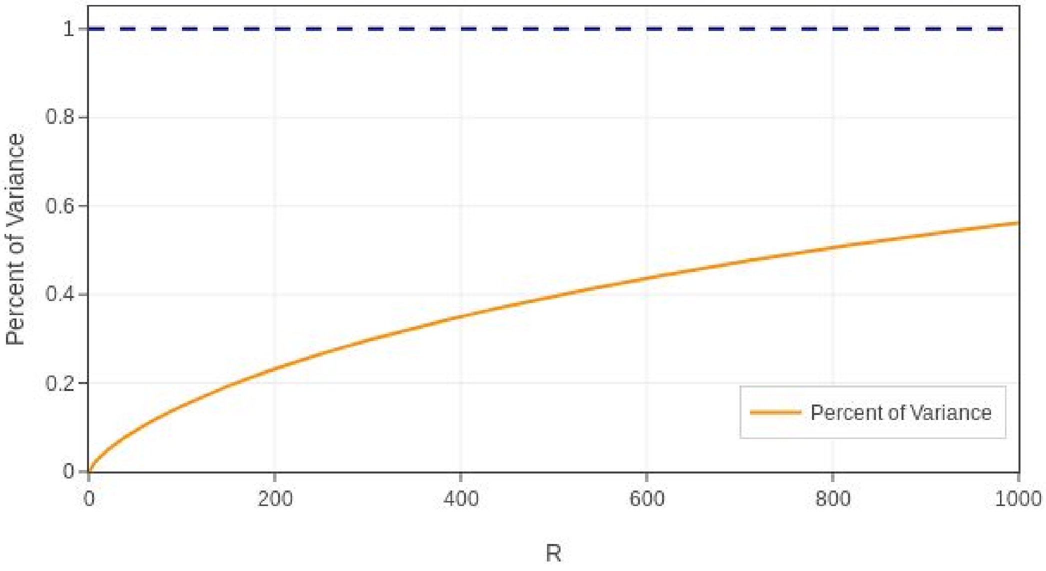

Figure 1: The Percent of Variance the Top r Principal Components Can Retain for r = 1 to 1000

Question 5

**Let r be the dimension that we want to reduce the data to (i.e. n_components).
Try r = 1, 2, 3, 5, 10, 20, 50, 100, 300, and plot the 5 measure scores v.s. r for both SVD and NMF.
Report the best r choice for SVD and NMF respectively.**

In this part, we utilized two methods (SVD and NMF) to accomplish the dimensionality reduction. The original dimension of our data is 27768, so we tried to reduce it within 1000 and plotted the percent of variance the top r principal components can retain.

Note that when using SVD to reduce the data to different dimensions, we didn't get "svd" matrix for each r using TruncatedSVD since selecting r most important dimensions will not change the values in the reduced matrix. Therefore, we get the 1000-d matrix and select the most important r features each time to construct the dimension reduced matrices with different r, and use them for K-Means model and record the 5 measures for each matrix and get the plot as Figure 2 below. Then, we compare the V-measure of all the r's and find the r corresponding to the best V-measure. The highest V-measure is when r = 2, since V-measure is defined to be the harmonic average of homogeneity score and completeness score and is a good representation of the overall performance of K-means. Therefore, we can conclude that r = 2 for SVD will maximize the K-means clustering performance on this dataset.

When using NMF to reduced the data to different dimensions, however, we should apply the NMF algorithm and get an r-dimension matrix for each r since the computation procedures of SVD and NMF are different. For example, we can't get the 1000-d matrix first and select most important 300 features, but have to construct a 300-d matrix and apply K-Means clustering on it.

After we calculate all the 5 measures for matrices with r from 1 to 300, we find that both SVD and NMF will result in highest values for 5 measures when the data is reduced to 2 dimensions (r = 2). Therefore, we can conclude that using r = 2 for SVD and NMF will help K-means model to get the best clustering result on this dataset.

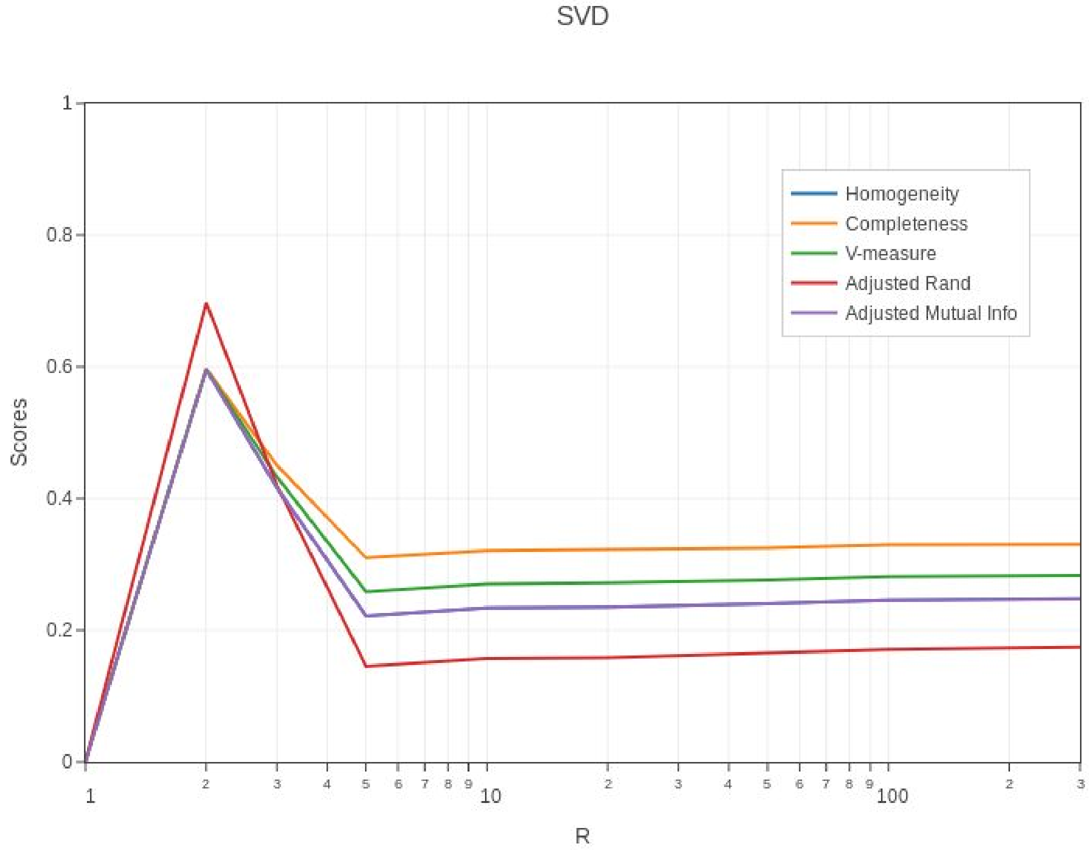

Figure 2: The 5 Measures scores of Data Reduced to r Dimension Using SVD

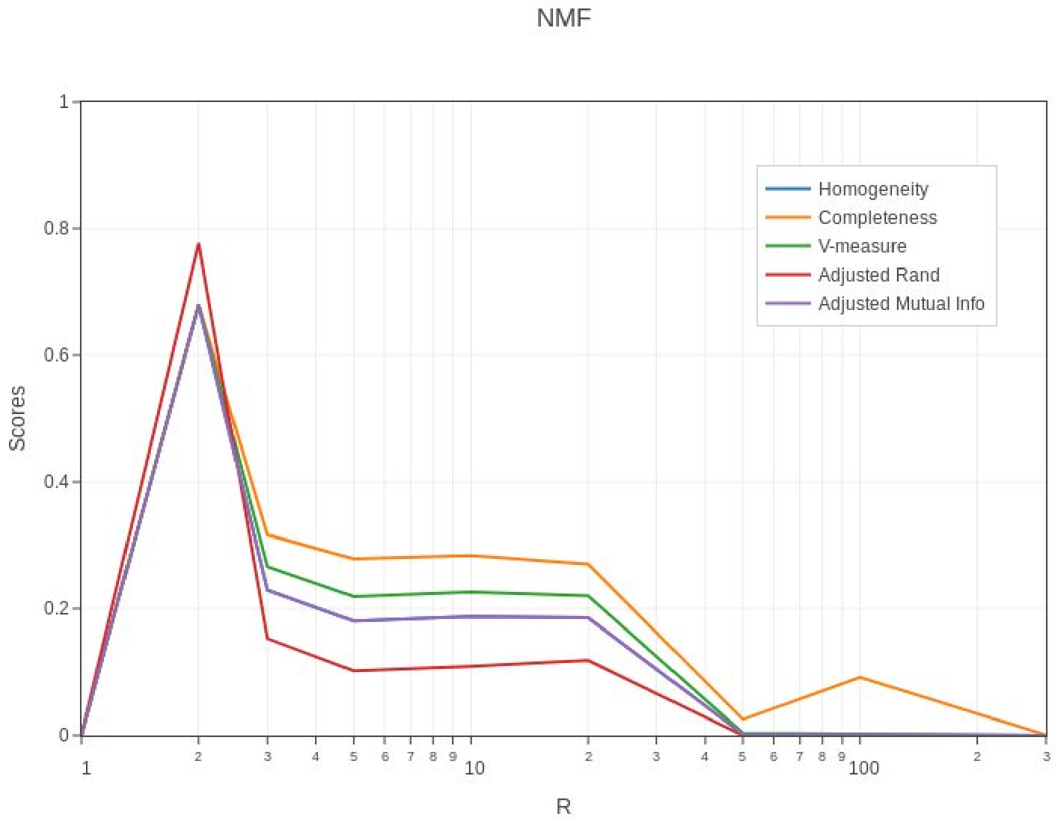

Figure 3: The 5 measures Scores of Data Reduced to r Dimension Using NMF

Question 6

How do you explain the non-monotonic behavior of the measures as r increases?

The reason of this non-monotonic behavior is when r is small, the vector can only keep a small amount of information (features) of the original data, so the clustering results will be poor; however when r is large, which means the data are in a high-dimensional space, the Euclidean distance is not a good feature for clustering anymore, and K-Means is an algorithm based on Euclidean distance, so the result will not be good when r is very large.

Normally, as r increases from 1 to a large number, the performance measures of K-Means clustering will first increase since more information are kept in the dimension-reduced data while the dimension is low and Euclidean distance is a good measure. But the performance measure will decrease when r gets larger to a high value, as K-Means doesn’t work well in high-dimensional space.

Question 7

Visualize the clustering results for:

- **SVD with its best r (r = 2)**

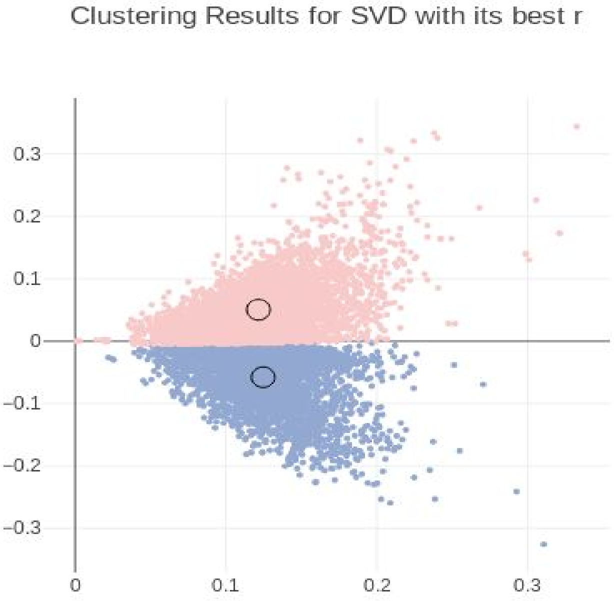

Figure 4: The Clustering Results Using SVD and r = 2

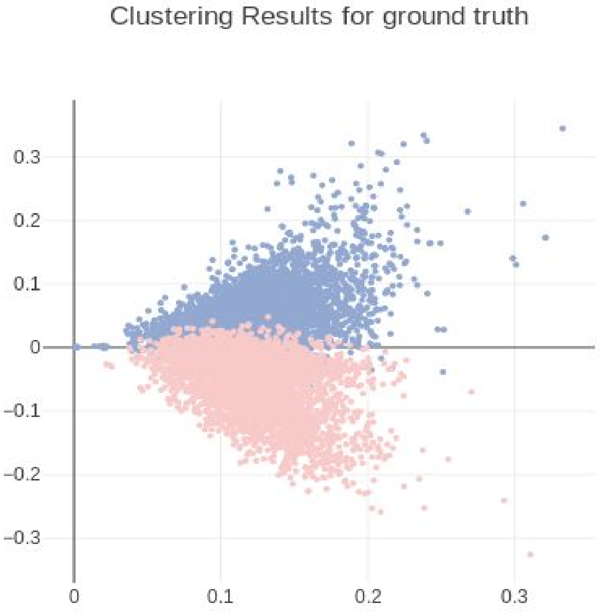

Figure 5: The Ground Truth Using SVD and r = 2

- **NMF with its best r (r = 2)**

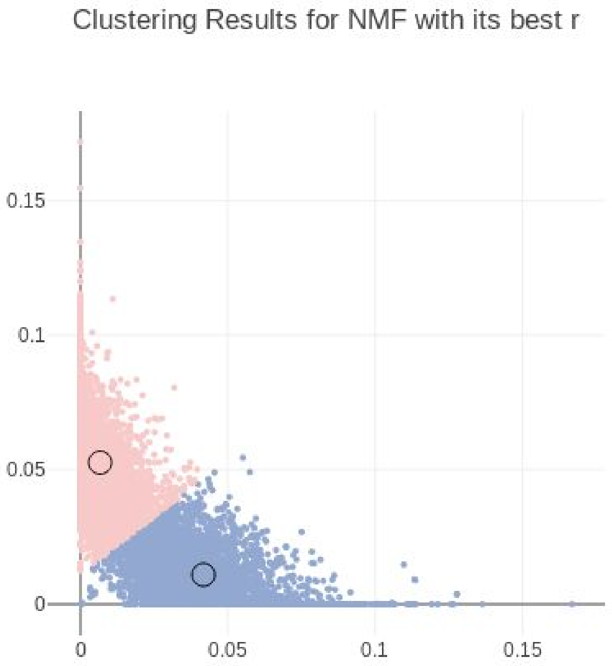
Clustering Results for NMF with its best r

Figure 6: The Clustering Results Using NMF and r = 2

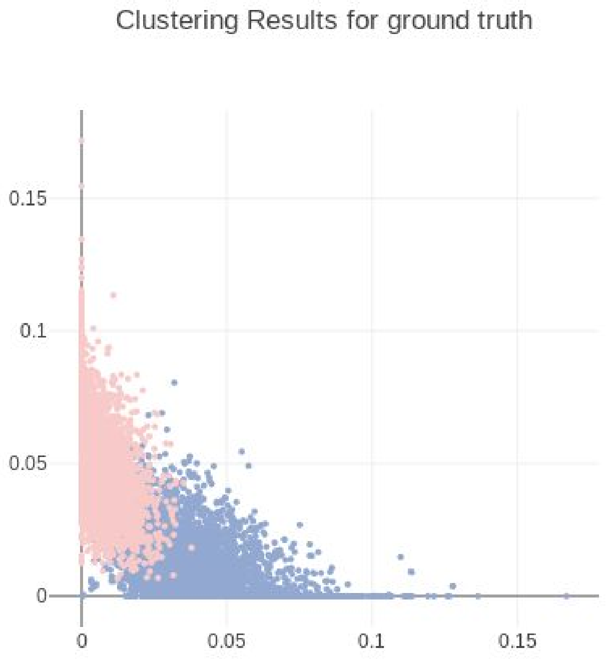
Clustering Results for ground truth

Figure 7: The Ground Truth Using NMF and r = 2

Question 8

Visualize the transformed data as in Question 7

We use sklearn.preprocessing.scale object for the unit variance transfomation, and use $f(x) = sign(x) \cdot (log(|x| + c) - log(c))$ as the non-linear transformation for both SVD and NMF.

- **SVD (r = 2) with unit variance**

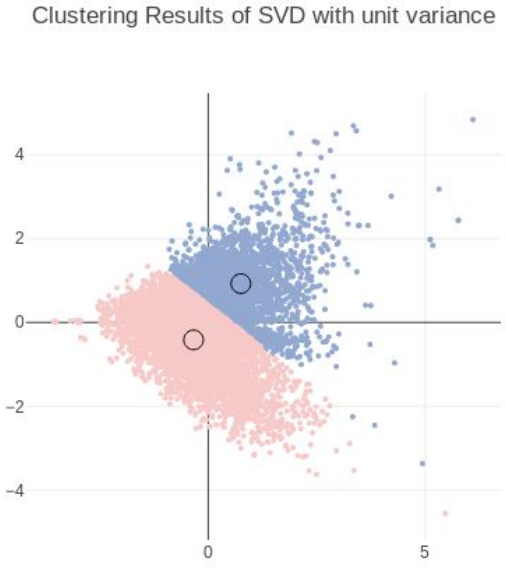

Figure 8: The Clustering Results Using SVD with Unit Variance Transformation

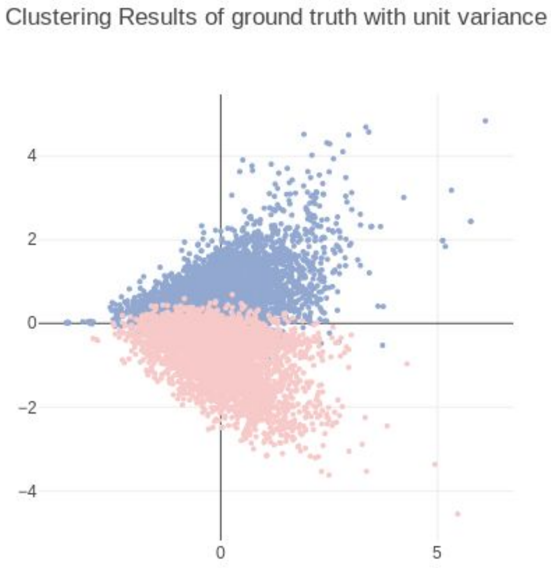

Figure 9: The Ground Truth Using SVD with Unit Variance Transformation

- **SVD (r = 2) with log transformation**

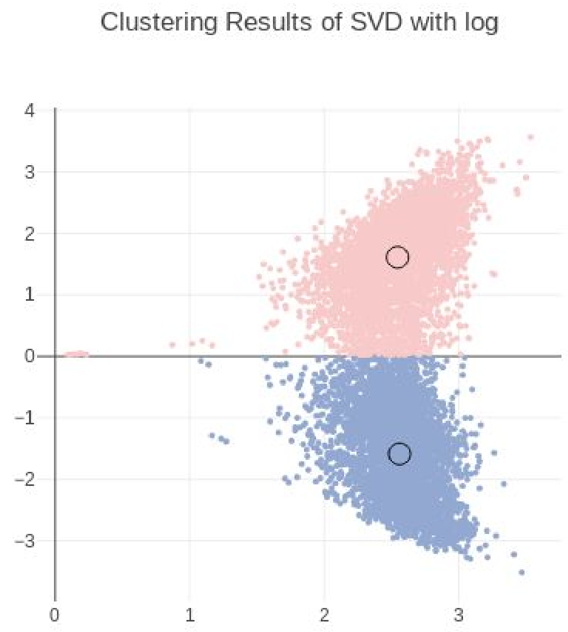

Figure 10 : The Clustering Results Using SVD with Log Transformation

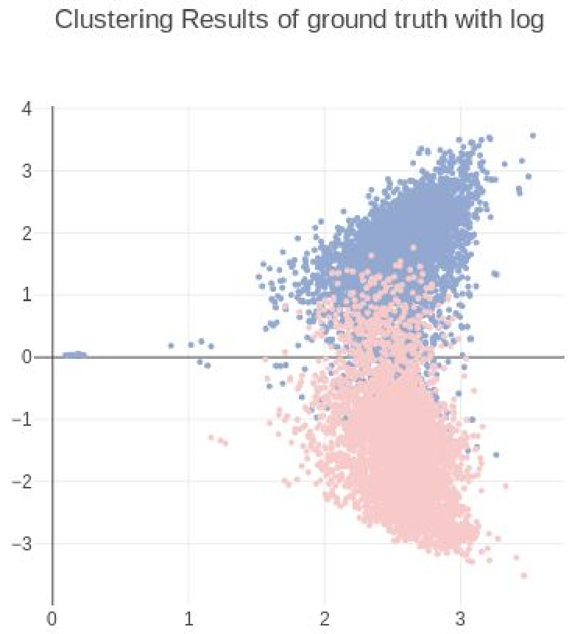

Figure 11 : The Ground Truth Using SVD with Log Transformation

- **SVD (r = 2) with unit variance transformation, then log transformation**

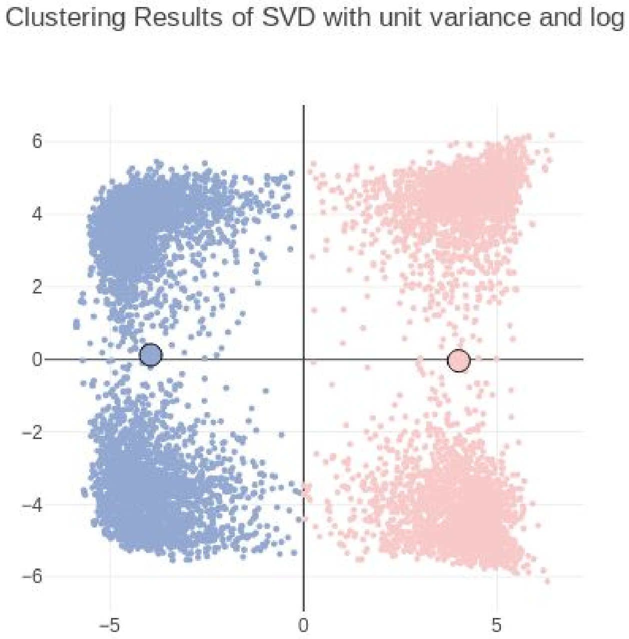

Figure 12 : The Clustering Results Using SVD with Unit Variance Transformation, then Log Transformation

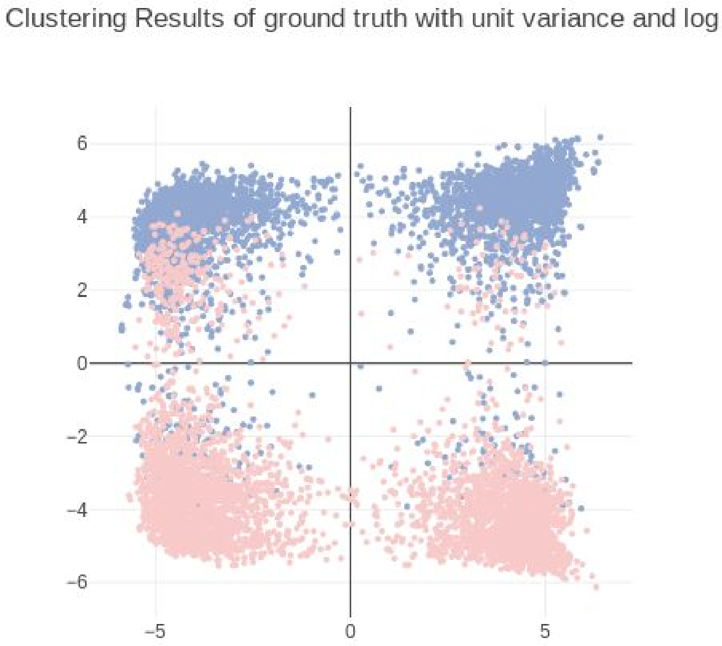

Figure 13 : The Ground Truth Using SVD with Unit Variance Transformation, then Log Transformation

• **SVD (r = 2) with log transformation, then unit variance transformation**

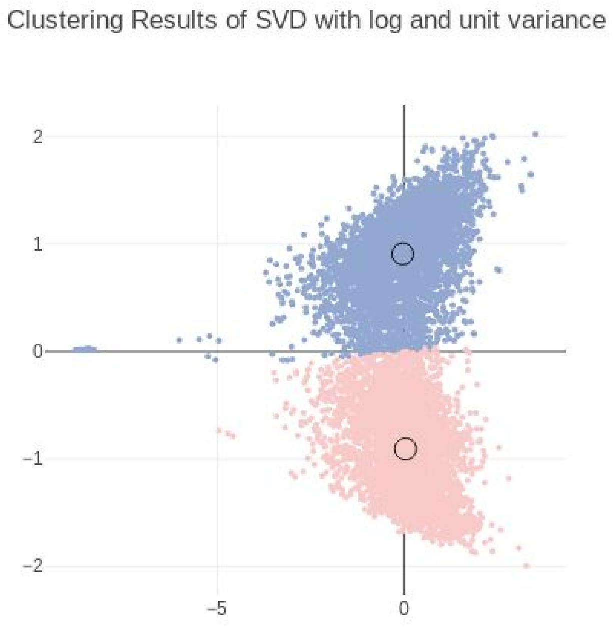

Figure 14: The Clustering Results Using SVD with Log Transformation, then Unit Variance Transformation

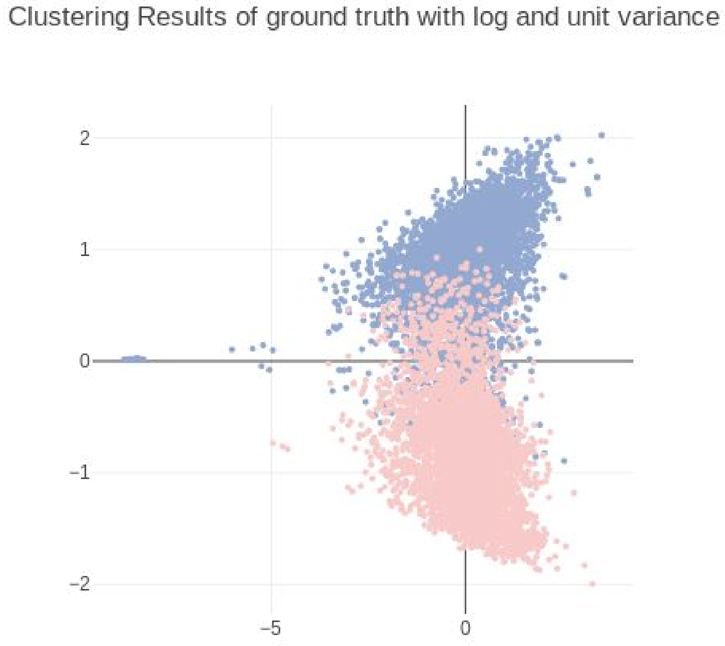

Figure 15: The Ground Truth Using SVD with Log Transformation, then Unit Variance Transformation

- **NMF (r = 2) with unit variance**

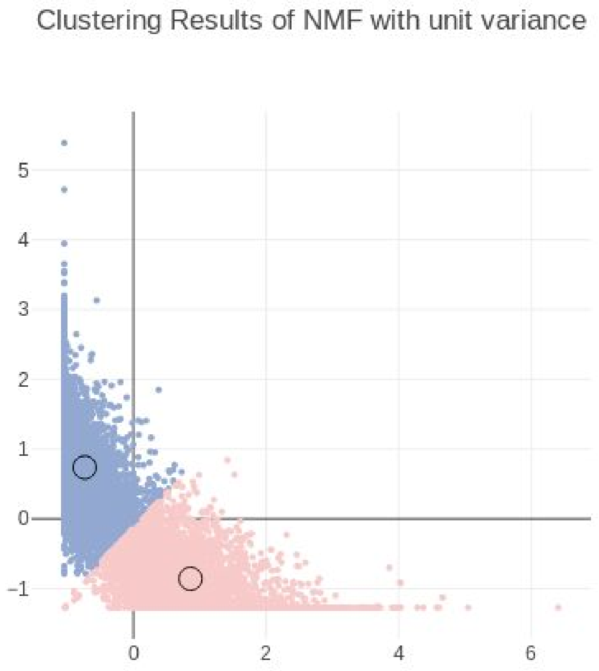

Figure 16: The Clustering Results Using NMF with Unit Variance Transformation

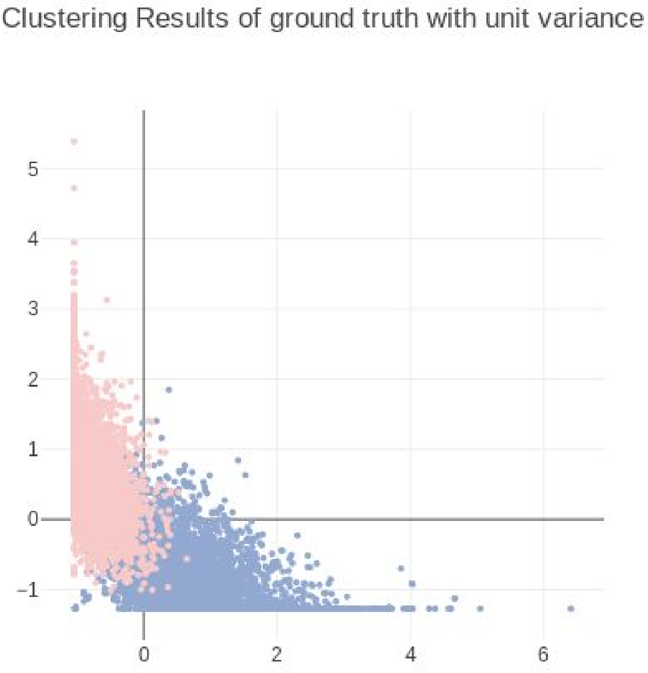

Figure 17: The Ground Truth Using NMF with Unit Variance Transformation

- **NMF (r = 2) with log transformation**

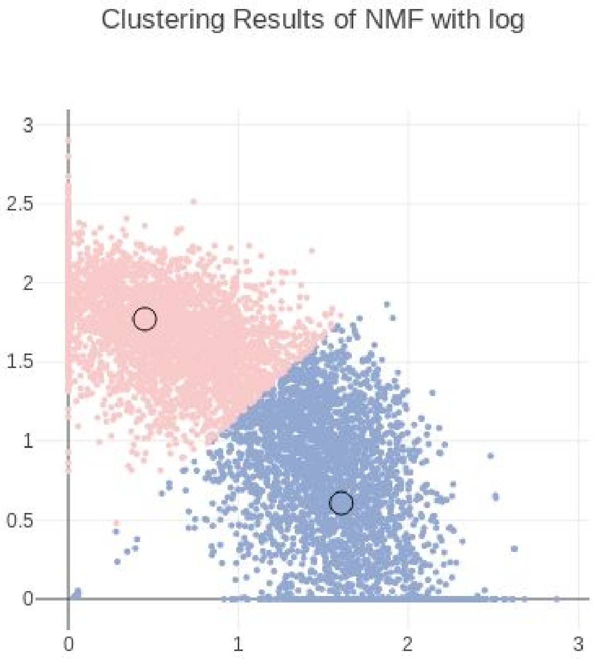

Figure 18: The Clustering Results Using NMF with Log Transformation

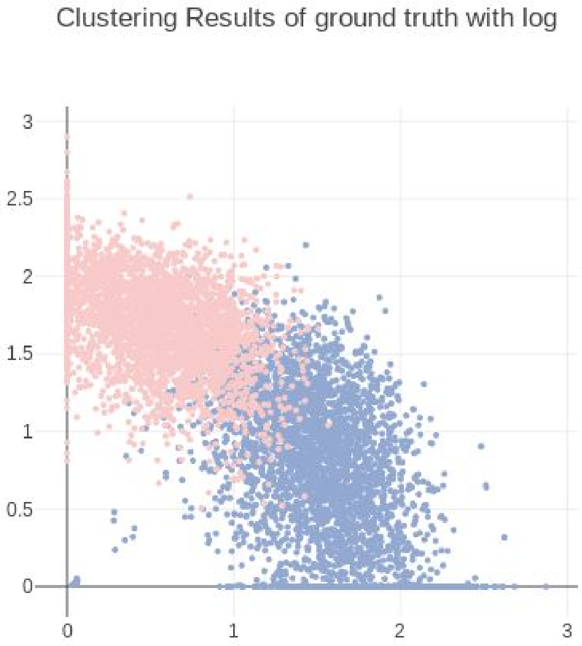

Figure 19: The Ground Truth Using NMF with Log Transformation

- **NMF (r = 2) with unit variance transformation, then log transformation**

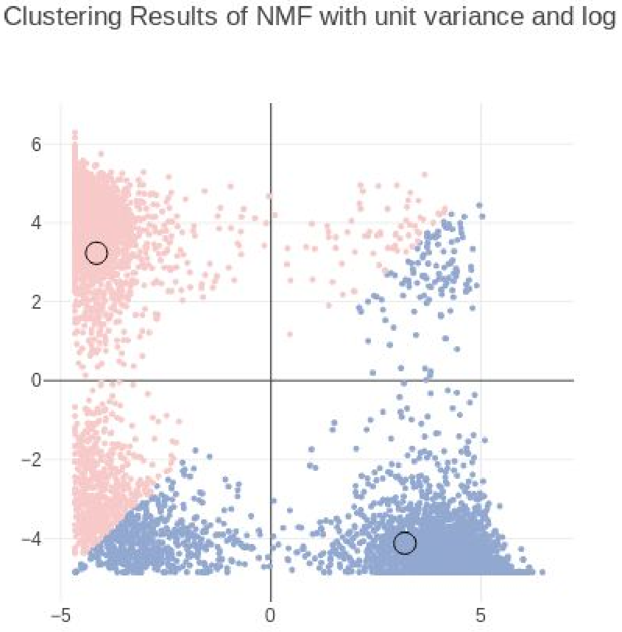

Figure 20 : The Clustering Results Using NMF with Unit Variance Transformation, then Log Transformation

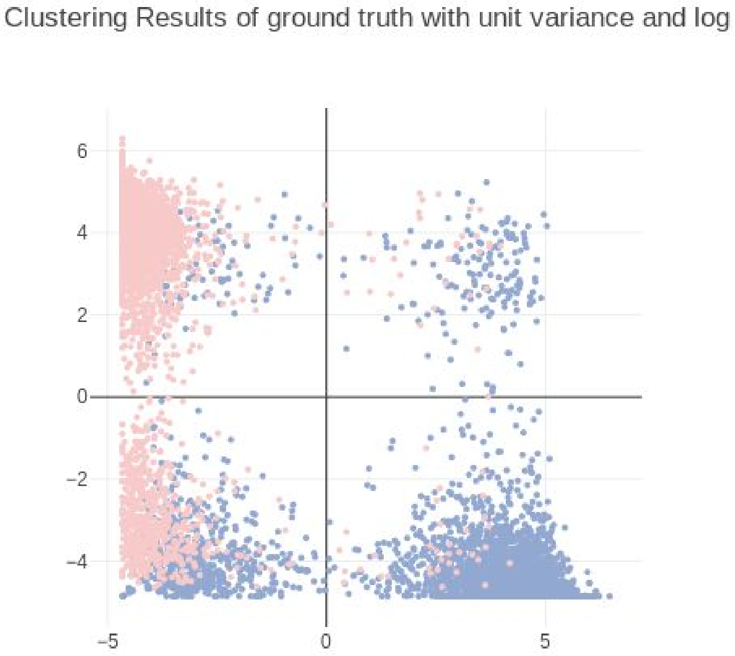

Figure 21 : The Ground Truth Using NMF with Unit Variance Transformation, then Log Transformation

- **NMF (r = 2) with log transformation, then unit variance transformation**

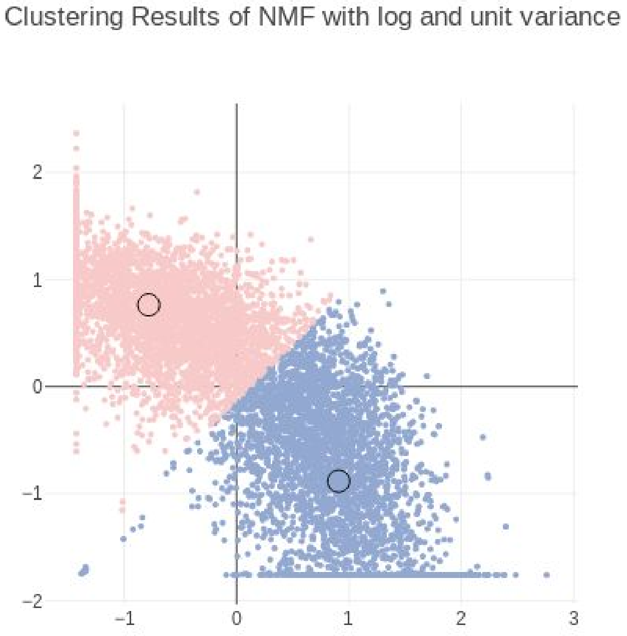

Figure 22: The Clustering Results Using NMF with Log Transformation, then Unit Variance Transformation

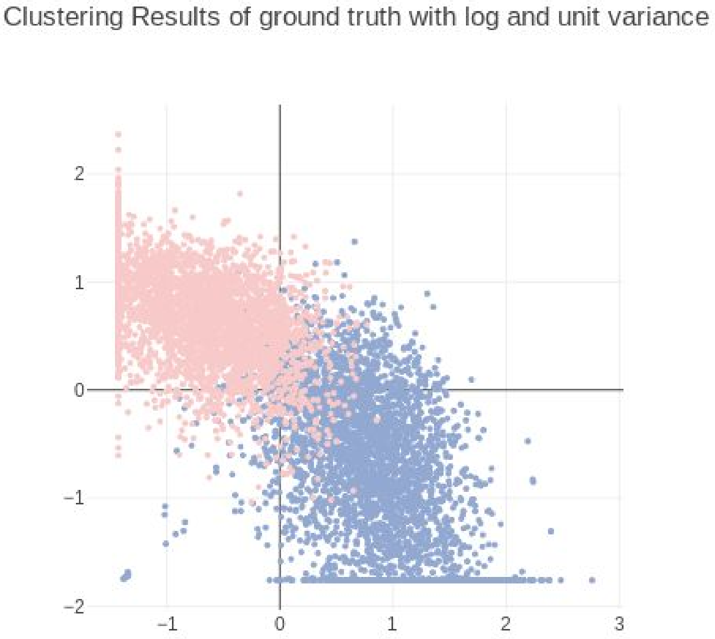

Figure 23: The Ground Truth Using NMF with Log Transformation, then Unit Variance Transformation

Question 9

Can you justify why the "logarithm transformation" may improve the clustering results?
The reason why "logarithm transformation" may improve the clustering results is that logarithm transformation decreases the difference between data points in certain dimension. It decreases the variability of data and makes the data conform more closely to normal distribution. Therefore, after logarithm transformation, we are likely to get better results.

Question 10

Report the new clustering measures (except for the contingency matrix) for the clustering results of the transformed data.

	Homogeneity	**Completeness**	**V-measure**	**Adjusted Rand-Index**	**Adjusted Mutual Information Score**
SVD + Unit Variance	0.235705	0.264157	0.249121	0.255138	0.235635
SVD + Log	0.610735	0.610713	0.610724	0.717791	0.610677
SVD + Unit Variance + Log	0.000074	0.000074	0.000074	-0.000013	-0.000017
SVD + Log + Unit Variance	0.610485	0.610444	0.610464	0.717362	0.610498

Table 2. 5 Measures for the K-means clustering with K = 2 Using SVD and Transformations

	Homogeneity	**Completeness**	**V-measure**	**Adjusted Rand-Index**	**Adjusted Mutual Information Score**
NMF + Unit Variance	0.682804	0.685646	0.684222	0.773443	0.682775

NMF + Log	0.675703	0.679139	0.677417	0.764985	0.675674
NMF + Unit Variance + Log	0.695575	0.696256	0.695915	0.792756	0.695547
NMF + Log + Unit Variance	0.686351	0.689055	0.687701	0.777018	0.686323

Table 3. 5 Measures for the K-means Clustering with K = 2 Using NMF and Transformations

Based on the results we get as shown in Table 2, using SVD with log (non-linear) transform and performing unit variance transform after log transform on SVD will slightly improve the K-Means clustering result. Using unit variance transform after SVD, on the other hand, will decrease the performance of K-Means model. Using Log transform after Unit Variance on SVD will decrease the performance further.

In the case of NMF as shown in Table 3, all of the four transformations can slightly improve the clustering measures using NMF. From the data we find using NMF with unit variance transformation and then log (non-linear) transformation yields results better than other transformations on this dataset.

Question 11

Repeat the following for 20 categories using the same parameters as in 2-class case:
• Transform corpus to TF-IDF matrix;
• Directly perform K-means and report the 5 measures and the contingency matrix;

The dimension of TF-IDF matrix we get from ‘20 News group’ dataset is 18846 x 52295 (18846 documents with 52295 terms) using TfidfVectorizer with min_df = 3. All data in 20 original sub-classes are retrieved.

Measure	Score
Homogeneity:	0.359421
Completeness:	0.451112
V-measure:	0.400080
Adjusted Rand-Index:	0.136636
Adjusted Mutual Information Score:	0.357319

Table 4. 5 Measures for the K-means Clustering with K = 20 Using the TF-IDF Dataset

The contingency matrix of K-Means clustering is as follow:

```
[[ 57  40   0   1   5  84   0   0  83   1   0   0   2 401  36   9   0  80   0   0]
 [ 82   0   1  16   1   1   2   0 241   0   0   4   1   3 525   0   0   0   0  96]
 [ 33   0  18   2   0   0  11   0 126   0   2   2   0   0 206   0   0   0   0 585]
 [ 25   0 230   7   1   0   5   0 175   0   0   5   0   0 437   0   3   0   0  94]
 [ 25   0 103  10   0   0   1   0 372   0   0   3   0   1 437   0   0   0   0  11]
 [ 86   0   1  25   0   0   2   0 143   3   0   4   0   1 569   0   0   0   0 154]
 [  5   0  70   3  27   0   7   0 477   0   0  12   5   0 334   0  12   0   0  23]
 [ 18   0   0   7 568   0   1   0 210   0   0   5   3   0 164  12   0   0   0   2]
 [ 77   0   0  17 682   0   1   0 110   0   0  12   0   0  97   0   0   0   0   0]
 [  2   0   0   2   0   0   1   0 312   0   0   2   4   1 171   0 499   0   0   0]
 [  2   0   0   3   2   0   0   0 110   0   0  50   0   1  83   0 748   0   0   0]
 [ 49   0   0   3   0   0  33   0  93 543   0   0  17   3 206  34   0   1   0   9]
 [ 49   0   8  35  29   0   1   0 242   1  13   7   0   0 582   0   1   0   0  16]
 [ 19   0   0  18   0   1   4  77 251   0  14   1   2  38 560   3   0   0   0   2]
 [ 21   0   0 486   2   0 107   0 140   0   0   0   1   9 211   9   0   0   0   1]
 [ 14   1   0   3   1 355   0   0  57   0  19   0   0 439 103   4   0   0   0   1]
 [ 12   0   0   5   2   0   2   0 118   4   0   5  74   3 129 556   0   0   0   0]
 [  5   0   0   0   0   2   0   0 107   0   0  18   0  72  84 238   0   0 414   0]
 [ 11   0   0  12   1   2   1   0 149   2   0  20   5  51 161 230   0 130   0   0]
 [ 14  71   0   0   2  93   4   2  81   0   0  13   1 195  85  57   0  10   0   0]]
```

Below is a diagram of contingency matrix. The deeper color refers to higher values in the contingency matrix.

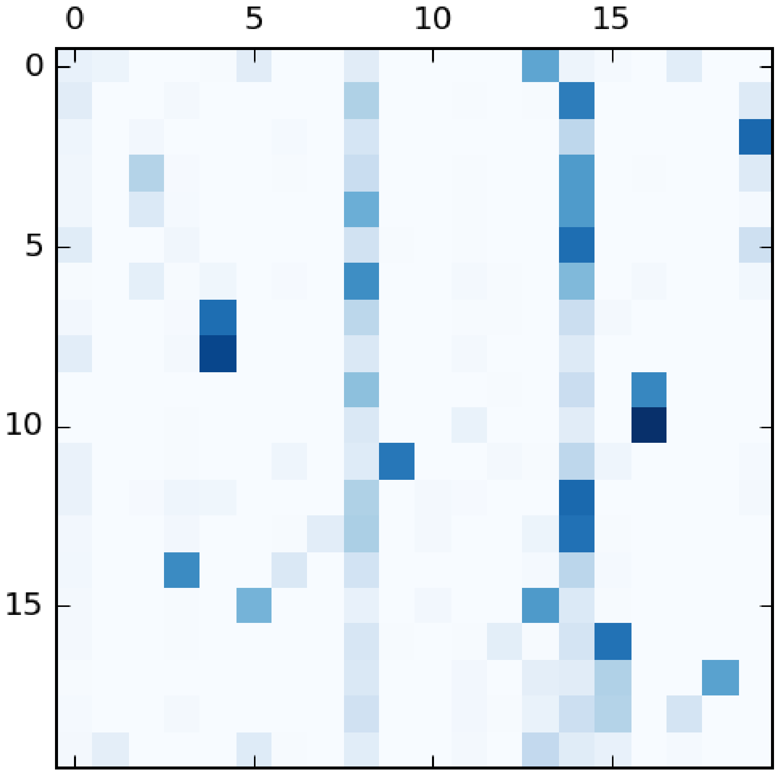

Figure 24: Contingency Matrix of K-Means Clustersing Directly Applied on 20 Catagories Dataset

Question 12

Try different dimensions for both truncated SVD and NMF dimensionality reduction techniques and the different transformations of the obtained feature vectors as outlined in above parts

You are asked, however, to report your best combination, and quantitatively report how much better it is compared to other combinations. You should also include typical combinations showing what choices are desirable (or undesirable).

In this question, our team expand the dimension r that the data will be reduced to a wider range of values. We experiments with r = 1, 2, 3, 5, 7, 9, 10, 11, 13, 16, 18, 20, 30, 40, 50, 60, 70, 80, 90, 100, 150, 200, 250, 300 and select the best combinations of reduced dimension and transformations.

For SVD, without any transformation, the 5 measures for K-Means clustering with different values are shown in the diagram below.

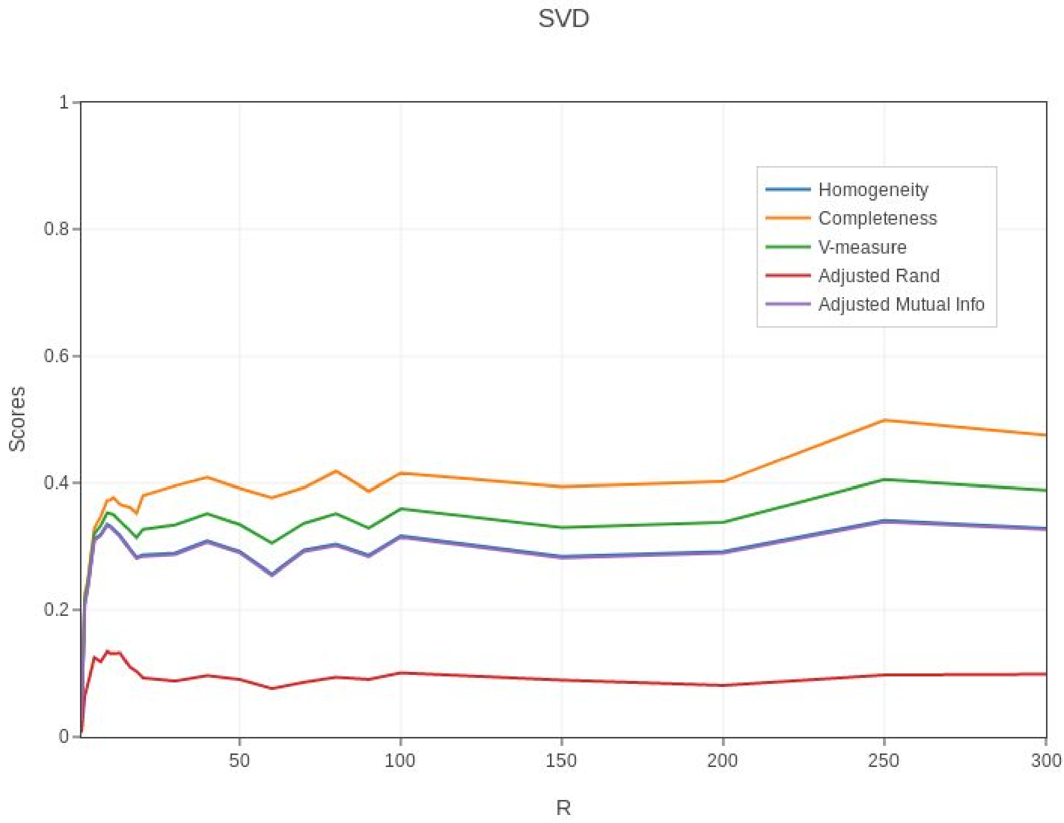

Figure 25: 5 Measures of K-Means Clustering on 20 Catagories Dataset Using SVD with Different r

We find when r = 250, the overall result of 5 measures is the best when no transformation applied with SVD.

For NMF, without any transformation, the 5 measures for K-Means clustering with different values are shown in the diagram below.

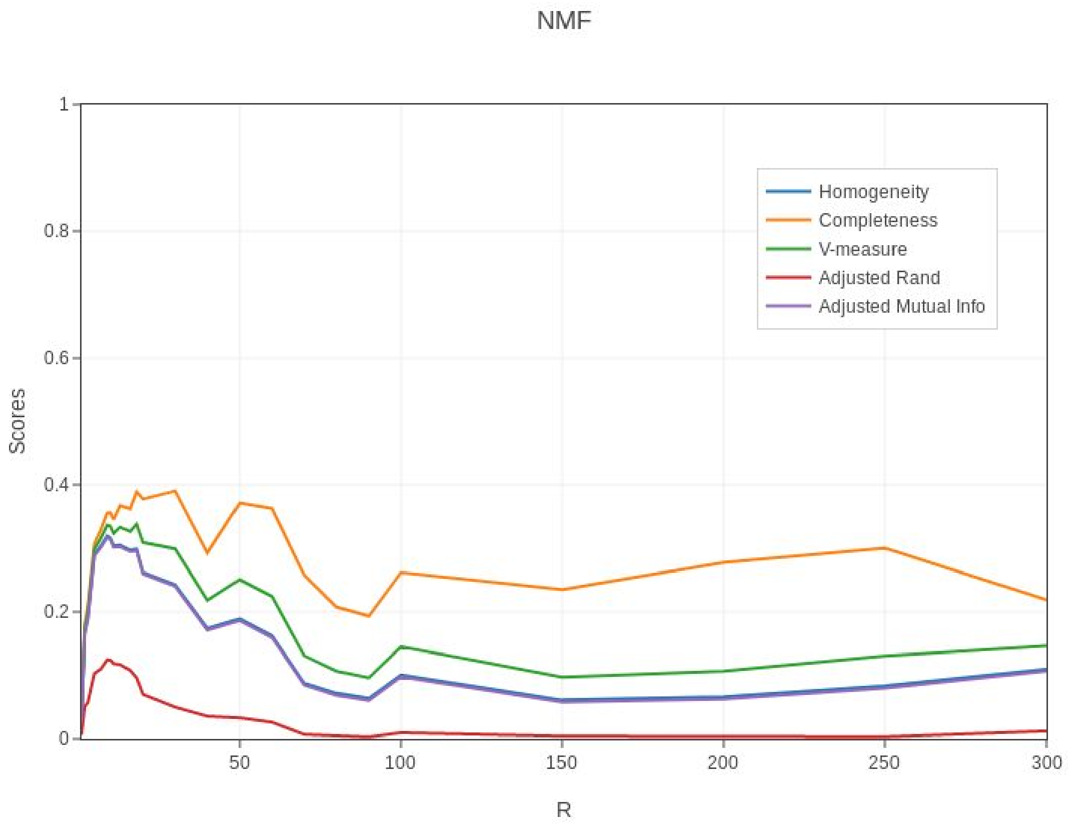

Figure 26: 5 Measures of K-Means Clustering on 20 Catagories Dataset Using NMF with Different r

We find when r = 18, the overall result of 5 measures is the best when no transformation applied with NMF.

Then we experiment with different transformation on SVD and NMF. We use the V-measure of K-Means clustering as the standard to compare the clustering result of SVD and NMF after applying different transformation for simplicity.

For SVD, the V-measures for K-Means clustering of different dimension and transformation combinations are shown in the diagram below.

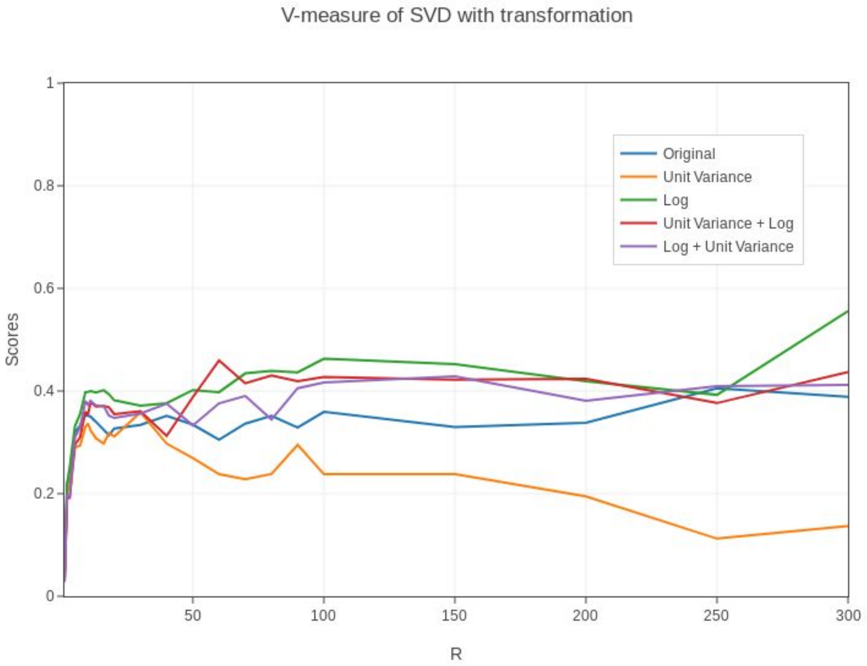

Figure 27: The V-measures for K-Means clustering of different dimension and transformation combinations using SVD

For NMF, the V-measures for K-Means clustering of different dimension and transformation combinations are shown in the diagram below.

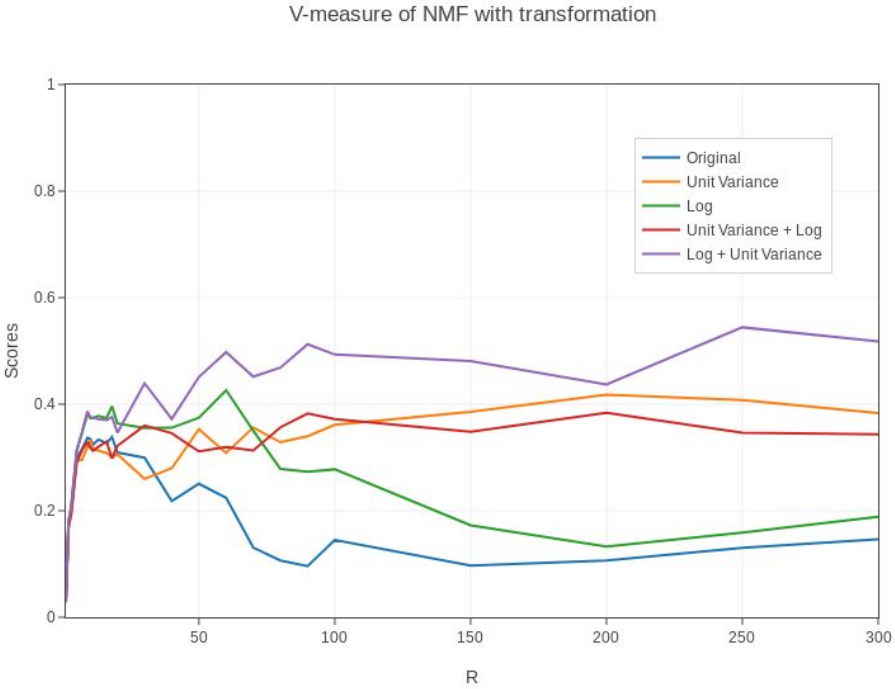

Figure 28: The V-measures for K-Means clustering of different dimension and transformation combinations using NMF

From the diagram we find the best combination for SVD is r = 300 with log transformation, and for NMF is r = 250 with log transformation then unit variance transformation. To show how the best combinations of reduced dimension and transformation metric improve the clustering result, we record the 5 measures of the best clustering result when no transformation applied and the 5 measures of the best combination setting.

	Homogeneity	**Completeness**	**V-measure**	**Adjusted Rand-Index**	**Adjusted Mutual Information Score**
SVD without transformation (r = 250)	0.291527	0.406106	0.339407	0.085645	0.289189
SVD with log transformation (r = 300) (best combination)	0.429919	0.506616	0.465127	0.219586	0.428056

Table 5. 5 Measures for the K-means Clustering Using Different Combination Settings on SVD

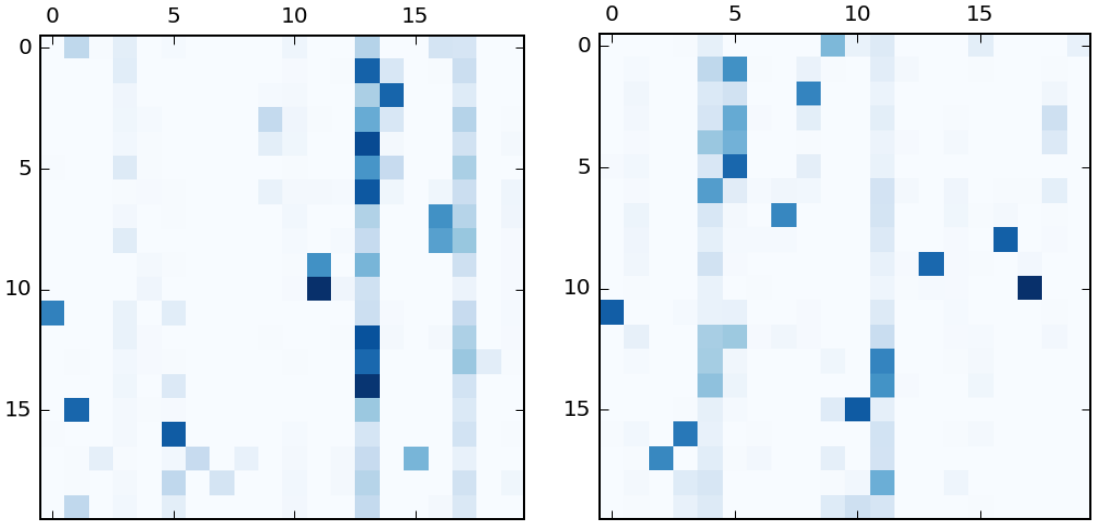

Figure 29: (Left) Contingency Matrix of SVD with no transformation, r = 250
(Right) Contingency Matrix of SVD with log transformation, r = 300

	Homogeneity	**Completeness**	**V-measure**	**Adjusted Rand-Index**	**Adjusted Mutual Information Score**
NMF without transformation (r = 18)	0.292205	0.359372	0.322326	0.105570	0.289893

NMF with log transformation, then unit variance transformation (r = 250) (best combination)	0.464747	0.550524	0.504012	0.204806	0.463000

Table 6. 5 Measures for the K-means Clustering Using Different Combination Settings on NMF

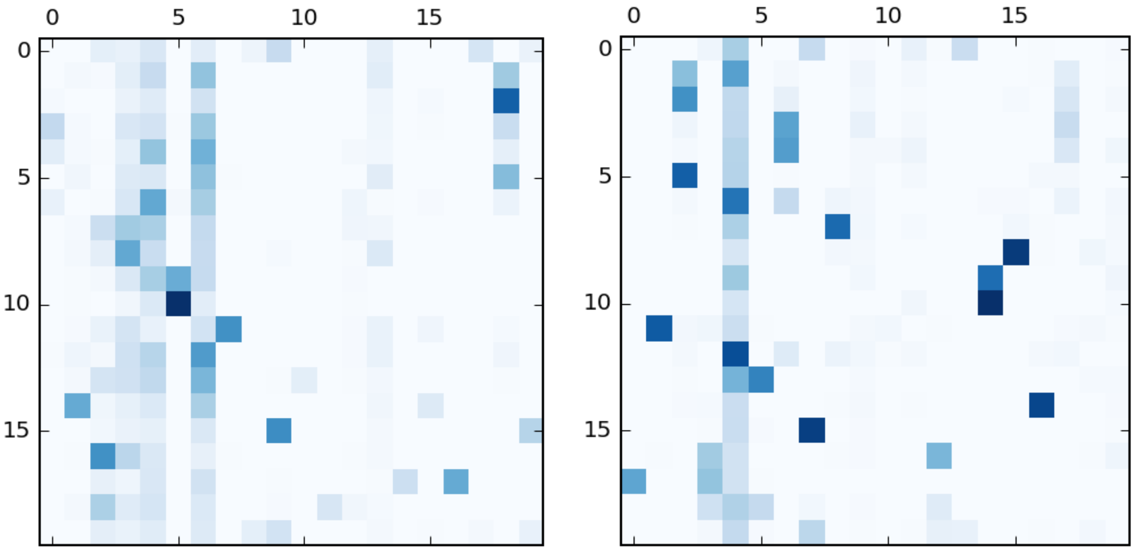

Figure 30: (Left) Contingency Matrix of NMF with no transformation, r = 18
(Right) Contingency Matrix of NMF with log transformation and unit variance transformation, r = 250

From the data in tables above, we can see when suitable transformation metric is applied on the data and when we reduce the data to a suitable dimension, the 5 measures for K-Means clustering can have significant improvement. We also find the best dimension of reduced data changes when transformation is added. For example, the best r for NMF is low (about 18) when no transformation added but is high (about 250) when log transformation and unit variance transformation applied.

Due to the randomness in computation process of SVD and NMF, the best combination of dimensions and transformation metrics may not be a fixed value and lie in a range of selections. From the diagram, SVD will give best performance when r is about 300 using log transformation, and NMF will give best performance when r is about 250 - 300 using log transformation and unit variance transformation. More experiments on combinations of dimensions and transformations should be carried out if time permits.

Reference

[1] Why is Euclidean distance not a good metric in high dimensions? [online]. (https://stats.stackexchange.com/questions/99171/why-is-euclidean-distance-not-a-good-metric-in-high-dimensions).

[2] Log-transformation and its implications for data analysis [online]. (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4120293/).