Project 2: Clustering

Zeyu Bai, 904514671 Yi-Jui Chang, 804587644
Ruizhi Yang, 704514506 Nan Liu, 004434221

Winter 2019

Introduction

In this report, we perform clustering on the "20 Newsgroups" dataset using K-means algorithm. The dataset includes 20 different newgroups which are nearly evenly distributed.

First consider a simple case which comprises two classes: computer and recreation. We pretend the class labels are unknown and apply the K-means clustering using the high dimensional sparse TF-IDF data. Since the Euclidean distance is not a good metric in the high dimensional space, the results show that K-means clustering does not yield a good result for the original data. To deal with this issue, we use the SVD and NMF to reduce the dimension of the data prior to applying the K-means clustering. Then the performance becomes better and also the computation is faster compared with the case without dimensional reduction.

Another problem for K-means algorithm is that it assumes the clusters are isotropically shaped, round-shaped. If not hold, then the algorithm might also fail to cluster properly. We then consider to perform the transformation on the reduced data by applying the scaling and logarithm transformation for the features. By comparing different cases, we find that applying log transformation and then scaling to the data performs the best among all the transformations methods.

Finally, we expand the dataset into 20 categories and repeat the procedure above, and also discuss the performance of the K-means clustering under different cases.

1 Dataset

The dataset we use is the "20 Newsgroups" dataset which includes 20 different groups. To get started, we first consider only two groups which include 8 categories.

Question 1: Building the TF-IDF matrix

To transform the documents into TF-IDF vectors, we run the CountVectorizer first to get total counts and then apply the TF-IDF transformation. The parameters we use are $min\_df = 3$ and $stop\_words = \ 'english'$.
And the Dimensions of TF-IDF matrix we get is: $(7882, 27768)$

2 K-means clustering

In this section, we apply K-means clustering with $k = 2$ using the high dimensional sparse TF-IDF data. Given the clustering results together with the ground truth labels from the data, we calculated the contingency table of the clustering result in Question 2. And to evaluate the performance of the K-means algorithm, we use five measures for the K-means clustering results and reported them in Question 3.

Question 2: Contingency table

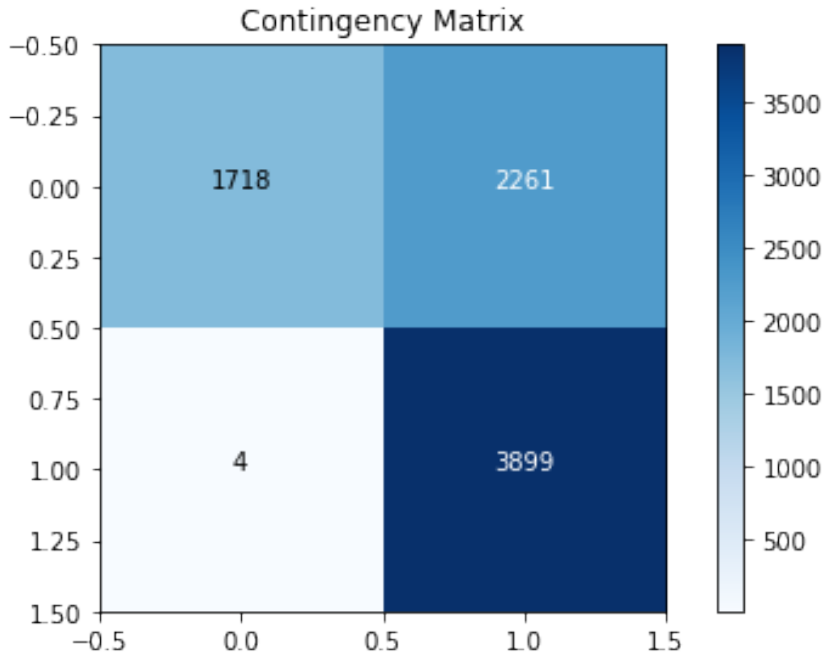


Figure 1: Contingency matrix of K-means with sparse data

Question 3: Evaluation measures

Table 1: Five Measures of K-means with sparse data

Homogeneity	0.254
Completeness	0.335
V-measure	0.289
Adjusted Rand-Index	0.181
Adjusted Mutual Infor. Score	0.254

From the contingency table in figure 1, the predicted label for one fo the groups contains both groups almost evenly. Thus the clustering does not have good performance which is also reflected in the 5 measures for the clustering results in table 1.

3 K-means clustering with dimension reduced data

In a high-dimensional space, the Euclidean distance is not a good metric anymore, which is commonly known as the "curse of dimensionality". Therefore, in this section, we perform the K-means clustering on the dimension reduced data from SVD or NMF.

Question 4: Percent of variance retained by SVD

We performed truncated SVD on the original data using different numbers of principal components. The percent of variance explained by the top r principal components of SVD is presented in figure 2. As r increases, the percentage of variance explained increases monotonically as expected.

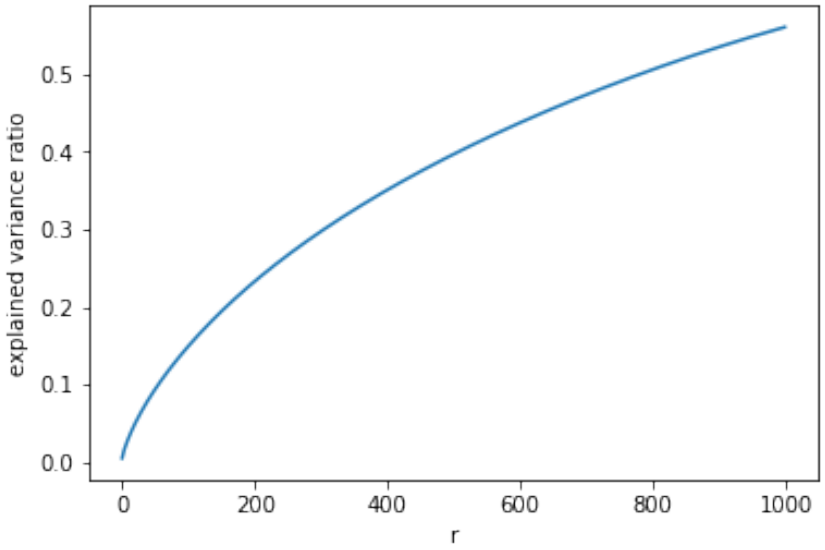


Figure 2: Percent of variance explained by the top r principal components

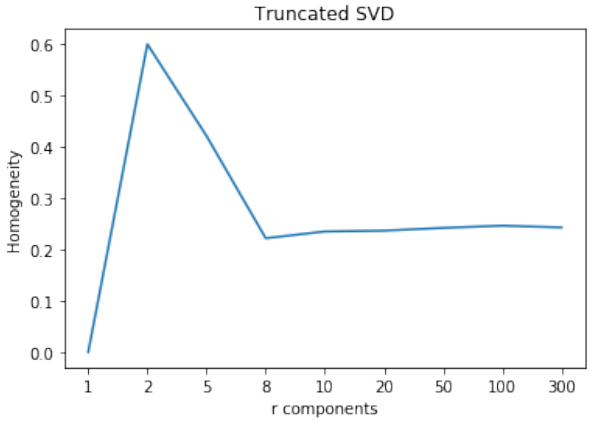
Question 5: Best r choice for SVD and NMF

In figure 3, we plotted 5 metrics of K-Means clustering using SVD with different values of r. The similar plots for K-Means clustering using NMF are given in figure 4. Comparing the curves for different metrics, we observe that these metrics have exactly the same tendency when r changes. It shows that in this 2 clustering task, all metrics can successfully measure the

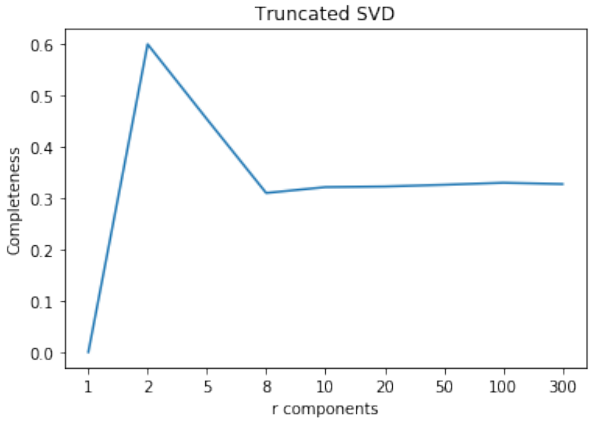
clustering accuracy. Therefore, we don't need to pick a specific measurement here. However, we value the V-measure, adjusted rand index, and adjusted mutual information more if some measures contradict with each other. The justification is that the homogeneity only measures the purity for the clustering, and the completeness only measures how complete each cluster covers the classes. On the other hand, V-measure is an harmonic mean of homogeneity and completeness which can be used as an overall measurement of the clustering. The adjusted rand index and the adjusted mutual information also measure of the similarity between two data clusterings.

Question 6: Explain the non-monotonic behavior of the measures as r increases

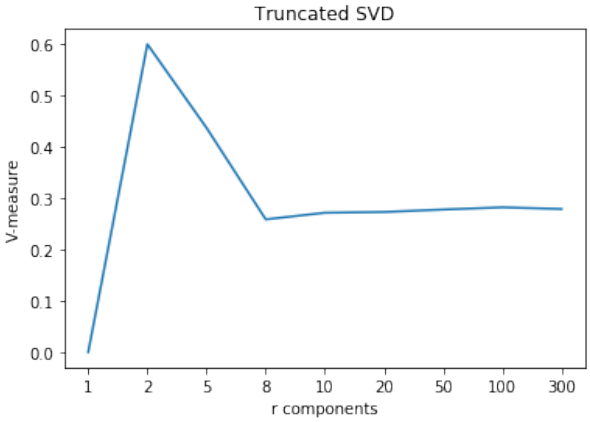
The measures for the clustering results are non-monotonic as the dimension of features(r) increases, and they all have the same pattern. For low dimensional features (r small), the measures are close to zero. As r goes up, the measures increase at first and then drop. The reason is that for low dimensional features, the information contained is very limited thus clustering can not be done properly. On the other hand if r is too large, we'll have high dimensional data. The Euclidean distance used in the K-means algorithm is not a good metric for high dimensional data, since there will be little difference in the Euclidean distances between different pairs of data points. Thus the K-means clustering for large r also does not perform good.



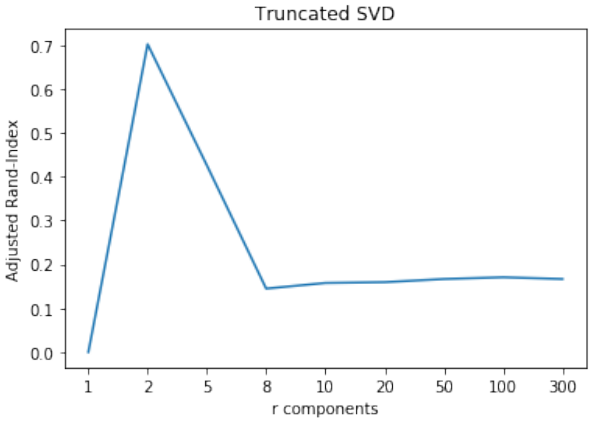
(a) Homogeneity



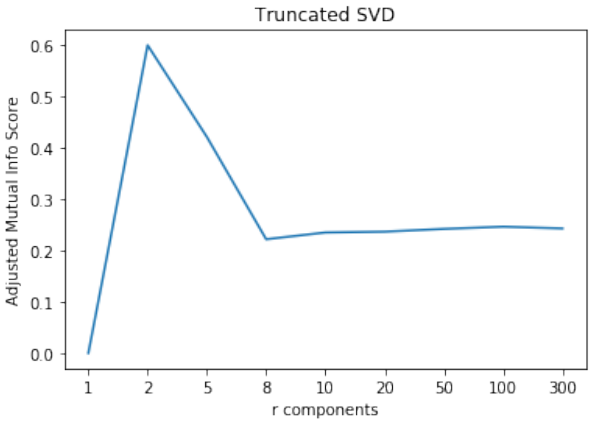
(b) Completeness



(c) V-measure

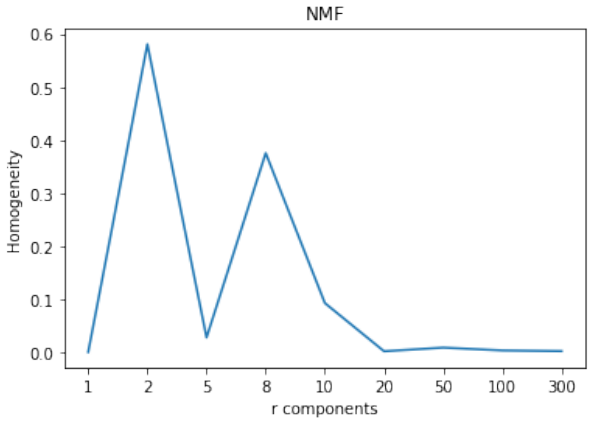


(d) Adjusted Rand-Index

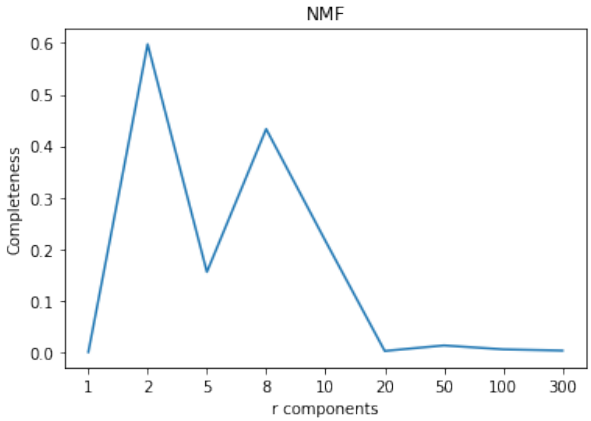


(e) Adjusted Mutual Infor. Score

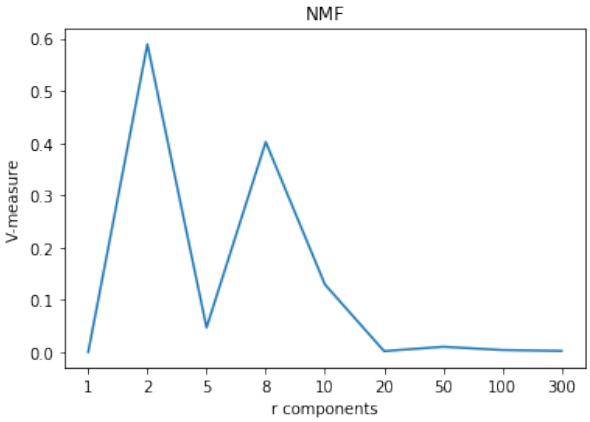
Figure 3: 5 measure scores versus r for SVD



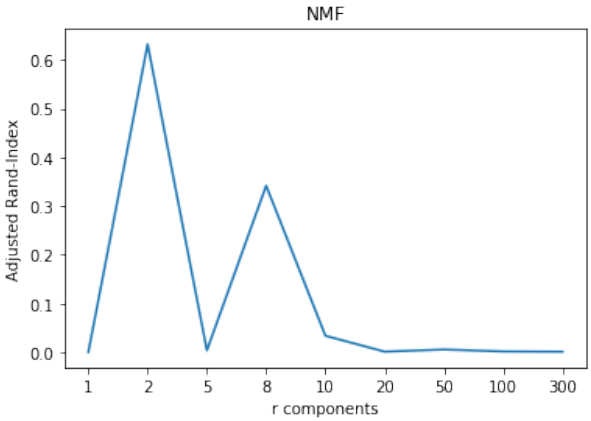
(a) Homogeneity



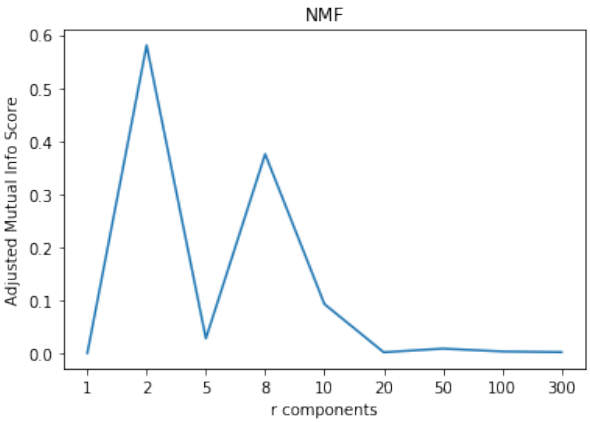
(b) Completeness



(c) V-measure



(d) Adjusted Rand-Index



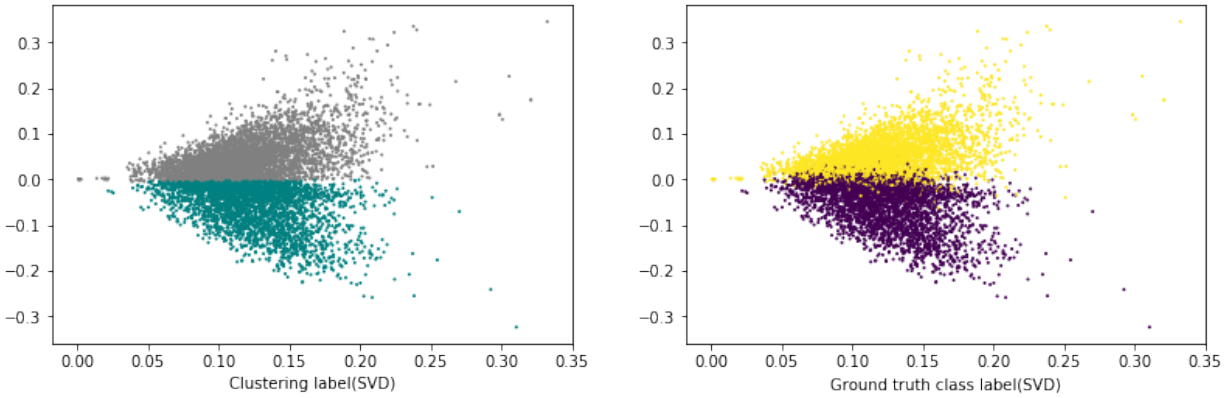
(e) Adjusted Mutual Infor. Score

Figure 4: 5 measure scores versus r for NMF

4 Visualization and transformation on reduced data

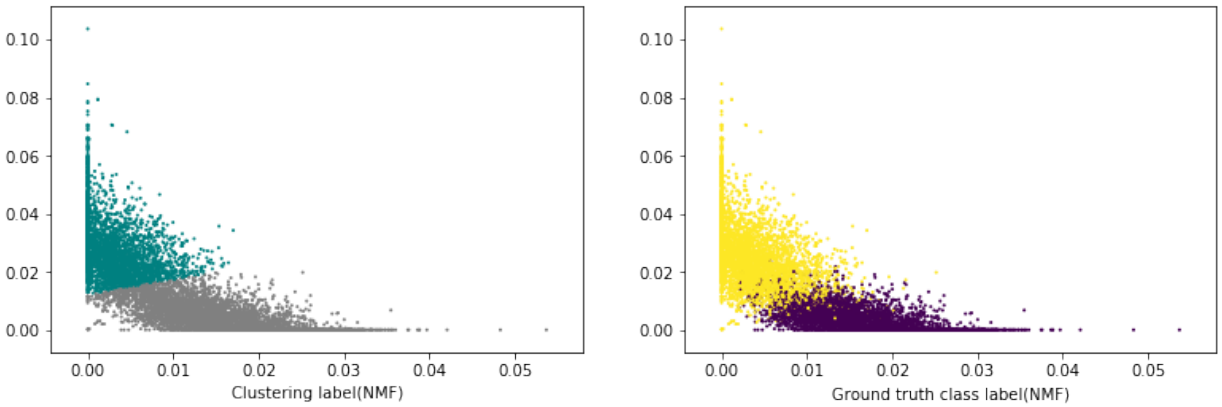
Question 7: Using PCA/NMF, K-Means vs ground truth results 2D visualization

Since we choose $r = 2$ based on 5 measurements in the previous section, reduced data with feature number 2 can be directly plotted here. Figure 5 shows visualization of truncated SVD(PCA), and figure 6 shows visualization of NMF. We use different colors for clustering plot and ground truth plot.



(a) (b)

Figure 5: Truncated SVD(PCA) with $r = 2$ visualized in 2D

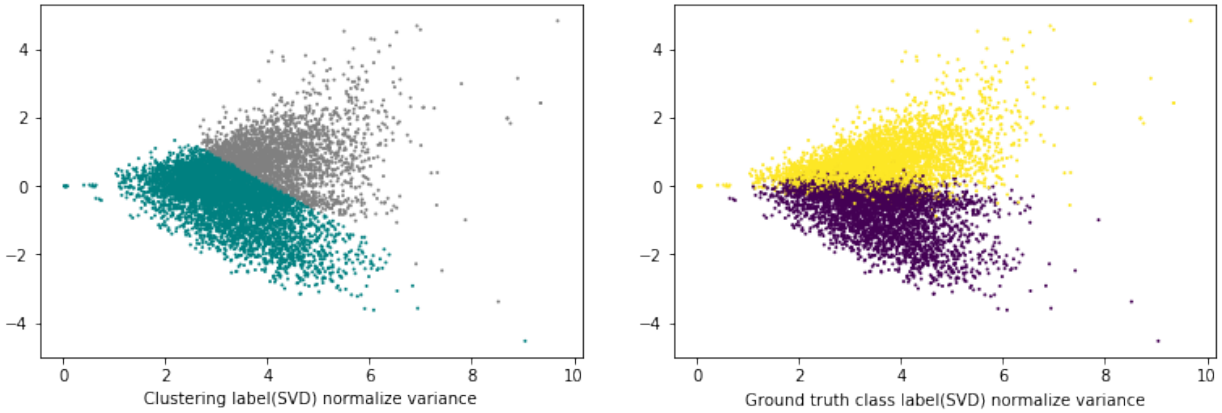


(a) (b)

Figure 6: NMF with $r = 2$ visualized in 2D

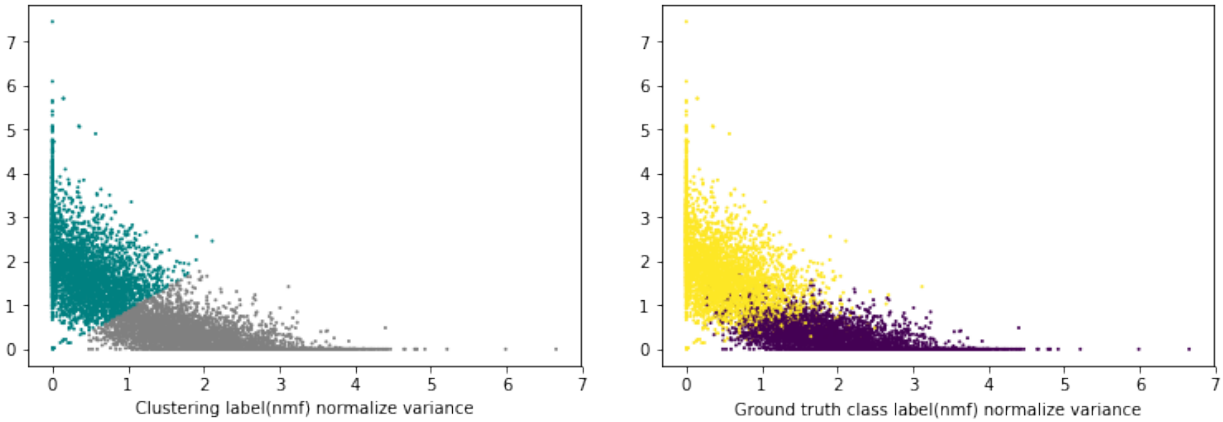
Question 8: Visualize the transformed data

A total amount of $2 \times (2 + 2)$ combination of feature reduction methods(PCA/NMF), feature scaling and logarithm transformation is done to perform 2 class clustering. Results are shown from figure 7 through figure 14.



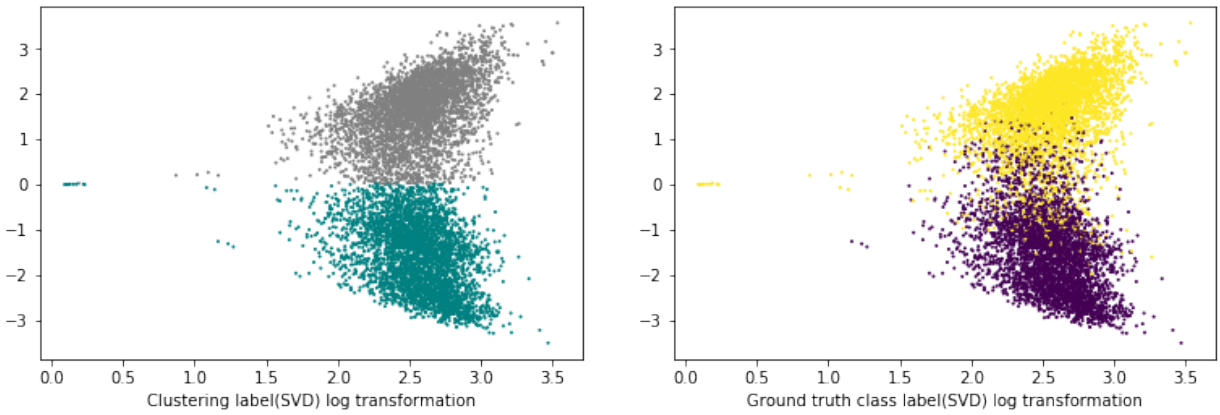
(a) (b)

Figure 7: Truncated SVD with $r = 2$, feature scaling



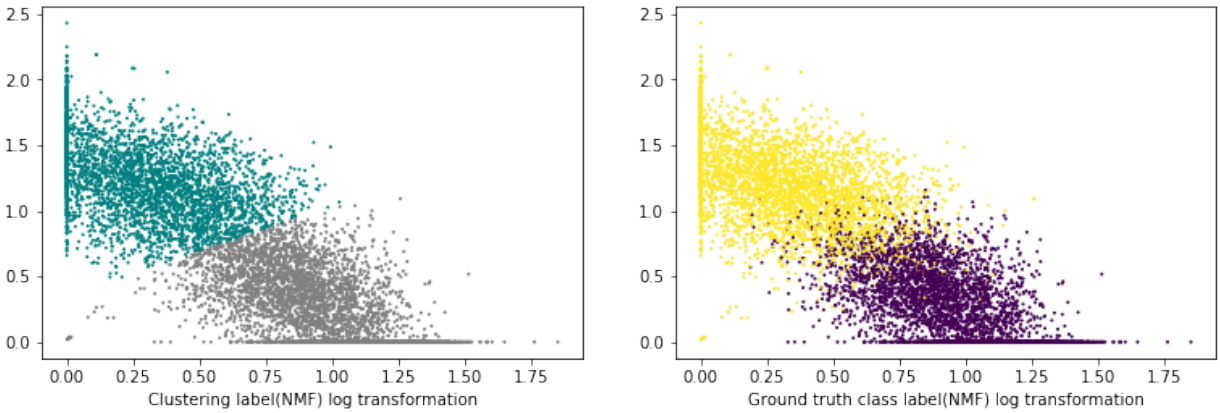
(a) (b)

Figure 8: NMF with $r = 2$, feature scaling



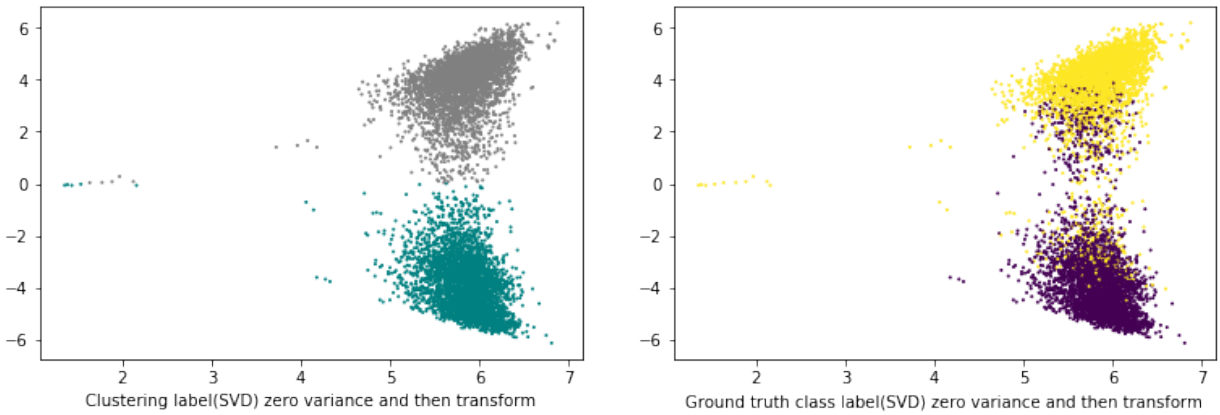
(a) (b)

Figure 9: Truncated SVD with $r = 2$, logarithm transformation



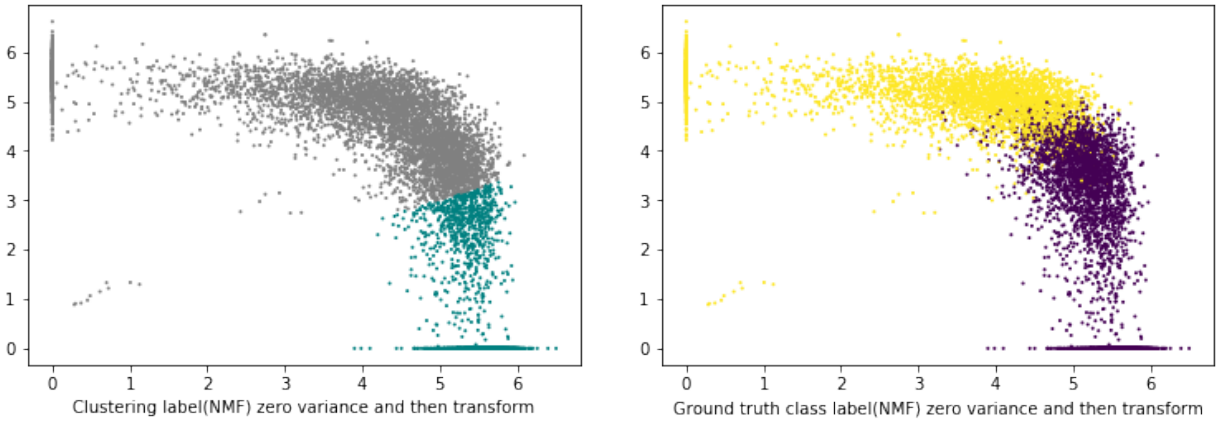
(a) (b)

Figure 10: NMF with $r = 2$, logarithm transformation



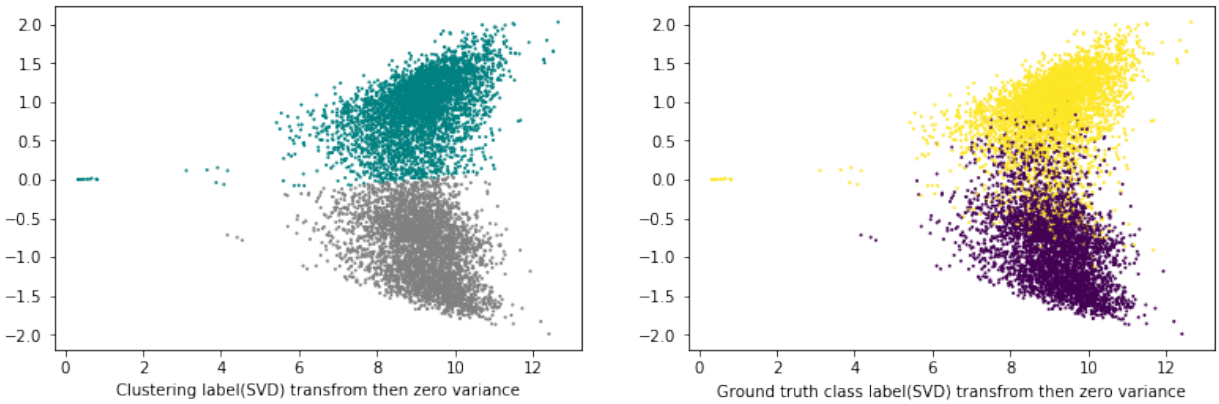
(a) (b)

Figure 11: Truncated SVD with $r = 2$, scaling first then log transformation



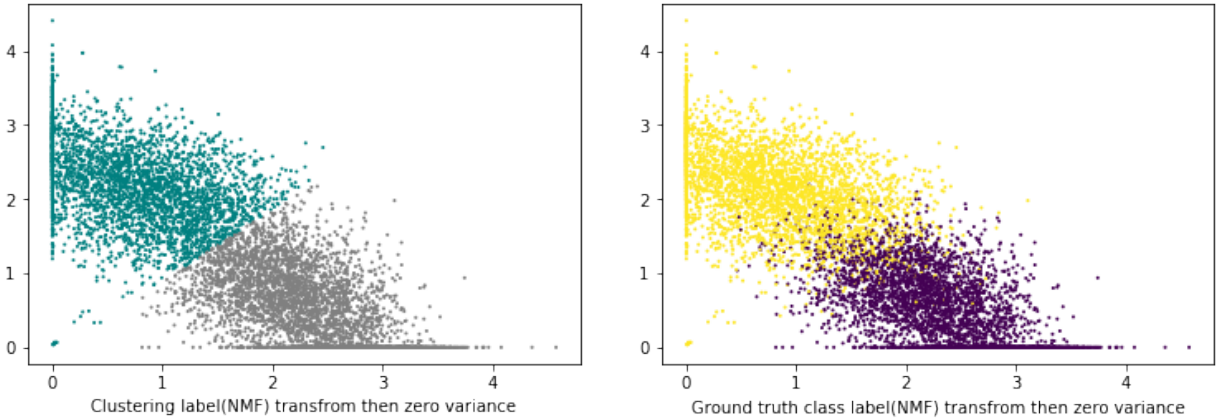
(a) (b)

Figure 12: NMF with $r = 2$, scaling first then log transformation



(a) (b)

Figure 13: Truncated SVD with $r = 2$, log transformation first then scaling



(a) (b)

Figure 14: NMF with $r = 2$, log transformation first then scaling

Question 9: Logarithm transformation

The K-means algorithm uses the Euclidean distance which implicitly assumes the clusters are isotropically shaped. And if the data is highly skewed then the performance of the clustering results can be very poor. Using logarithm transformation, the data points will spread more uniformly, meaning that data very small can be amplified, data very large can be shrinked. By smoothing the ourliers, logarithm transformation can improve the clustering results.

Question 10: Report the new clustering measures for the clustering results of the transformed data

Table 2: Five measures of scaling/transformation manipulation with SVD

Measures	Scaling	Logarithm Transformation	Scaling Transformation	Transformation Scaling
Homogeneity	0.235	0.602	0.604	0.606
Completeness	0.263	0.602	0.604	0.606
V-measure	0.248	0.602	0.604	0.606
Adjusted Rand-Index	0.255	0.710	0.711	0.713
Adjusted Mutual Infor. Score	0.235	0.602	0.604	0.605

Table 3: Five measures of scaling/transformation manipulation with NMF

Measures	Scaling	Logarithm Transformation	Scaling Transformation	Transformation Scaling
Homogeneity	0.683	0.701	0.313	0.715
Completeness	0.686	0.702	0.383	0.715
V-measure	0.684	0.701	0.344	0.715
Adjusted Rand-Index	0.773	0.796	0.249	0.811
Adjusted Mutual Infor. Score	0.683	0.701	0.313	0.715

Table 2 and Table 3 reported the evaluation of the K-means clustering of transformation on SVD-reduced data and NMF-reduced data, respectively. Based on the 5 measures, applying log transformation then scaling performs the best for both the SVD-reduced and NMF-reduced data. The reason is that taking log can transform the skewed data to be more likely to be normally distributed, and then applying the scaling on variable we can obtain unit variance for the features. This transformed data should be more likely to satisfies the isotropic assumption which is required for the better performance of K-means clustering.

5 Expand dataset into 20 categories

Question 11: Simple K-means with 20 categories

Table 4

Homogeneity	0.359
Completeness	0.451
V-measure	0.400
Adjusted Rand-Index	0.137
Adjusted Mutual Infor. Score	0.357

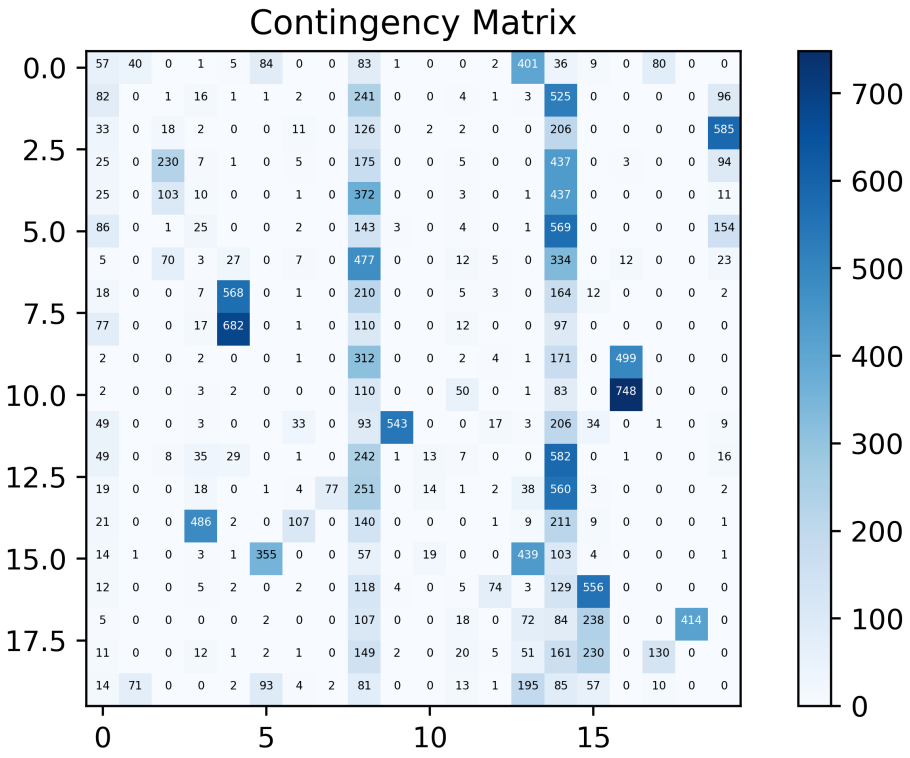


Figure 15: Contingency matrix of K-means with 20 clusters

Question 12: K-means with different r, dimension reduction method and transformation

At first we tried to determine the reduced dimension r, then apply the two types of transformation. This results in the choice of $r = 10$. However by applying different combination of transformations, all the clustering results are not ideal. We then decided to search for the best combination of r together with transformation types for the best performance. Best 5 combinations are listed based on V-measure, adjusted Rand Index and adjusted mutual information score respectively, from table 5 to table 7. All tables show that after applying transformation followed by scaling on the dimension reduced data using NMF ($r = 100$), the K-means clustering achieves the best accuracy. The contingency matrix is presented in figure 16. Comparing with figure 15, we can see that the number of blue squares decreases but their colors are deeper, which means the clustering are more similar with the ground truth. With the best combination of NMF with $r = 100$ transformation then scaling, 2D visualization of the clustering is given in figure 17.

Table 5: Top 5 Best V-measure

Rank	Score	SVD/NFM	r_{best}	Transformation type
1	0.538	NMF	100	transformation then scaling
2	0.497	NMF	300	transformation then scaling
3	0.449	SVD	100	transformation
4	0.445	SVD	300	transformation
5	0.431	SVD	300	transformation then scaling

Table 6: Top 5 Best Adjusted Rand Index

Rank	Score	SVD/NFM	r_{best}	Transformation type
1	0.275	NMF	100	transformation then scaling
2	0.265	NMF	50	scaling then transformation
3	0.233	SVD	100	transformation
4	0.230	SVD	10	scaling then transformation
5	0.228	NMF	300	scaling then transformation

Table 7: Top 5 Best Adjusted Mutual Information Score

Rank	Score	SVD/NFM	r_{best}	Transformation type
1	0.502	NMF	100	transformation then scaling
2	0.444	NMF	300	transformation then scaling
3	0.430	SVD	100	transformation
4	0.412	SVD	300	transformation
5	0.405	NMF	50	scaling then transformation

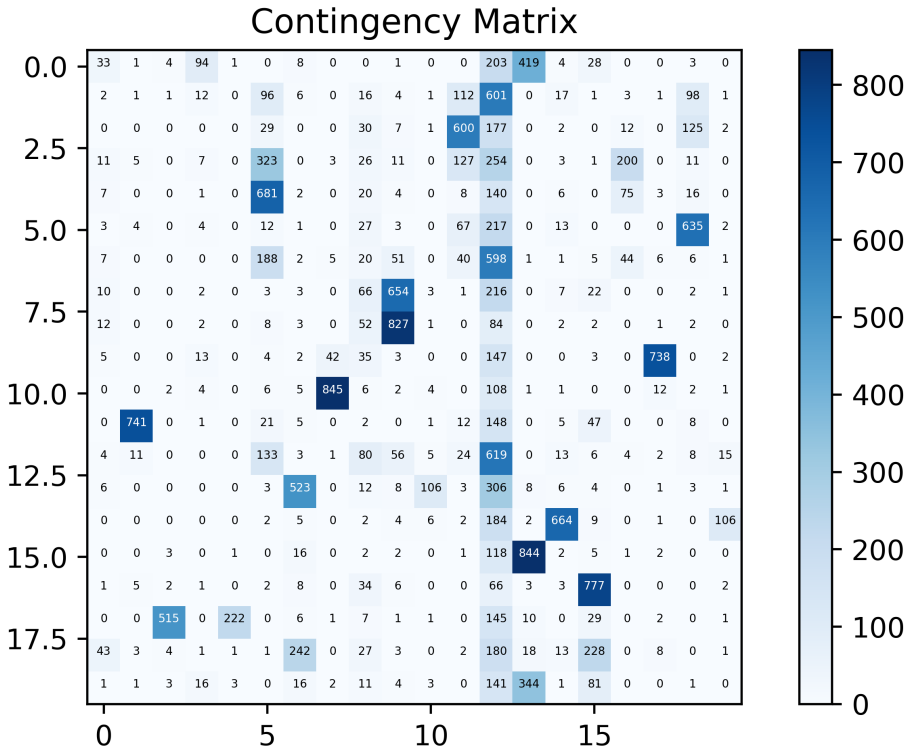
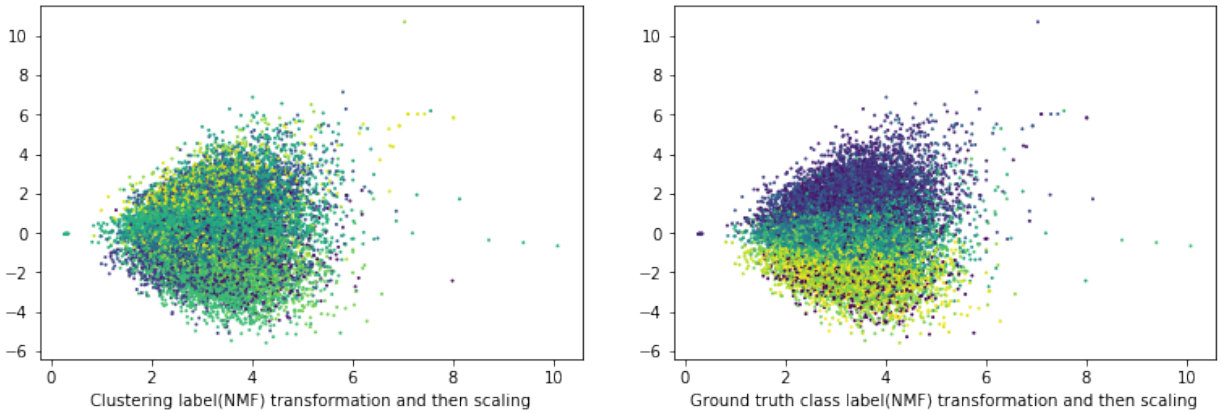


Figure 16: Contingency matrix of the best clustering for 20 clusters



(a) (b)

Figure 17: NMF with $r = 100$, log transformation then scaling