EE 219 Project 2

Yifan Shu, Chengshun Zhang, Xuan Yang

Introduction

Cluster algorithms are unsupervised methods for finding groups of data points that have similar representations in a feature space. Clustering differs from classification in that no a priori labeling (grouping) of the data points is available.

In this project, we were focusing on a cluster algorithm call K-mean clustering. K-means clustering is a simple and popular cluster algorithm. Given a set of data points $\{\vec{x}_1, ..., \vec{x}_N\}$ in multidimensional space, it tries to find K clusters s.t. each data point belongs to one and only one cluster, and the sum of the squares of the distances between each data point and the center of the cluster it belongs to is minimized.

During this project, we used the "20 Newsgroup" dataset. We firstly preprocessed the the textual data by tokenizing each document into word, and then transformed it into a document-item matrix. Then, we used TF-IDF score to finally determine our data.

With the transformed data, we explored the K-means algorithm and the evaluation method of a clustering result (contingency matrix, homogeneity score, completeness score, V-measure score, adjusted Rand Index score, and adjusted mutual information score). We also investigated how PCA, scaling, and logarithm transformation can affect the performance clustering result using K-means algorithm.

At last, we expanded our dataset from 2 categories to 20 categories. Using different combination of PCA, scaling, and logarithm transformation to get a best result.

Question 1

First, we had to transform the data into a matrix form, which can be easily handled. We followed the same method as in Project 1, we tokenizing each document into word using the min_df = 3 and excluding the 'English' stop words. The we used TF-IDF score to further calculate our TF-IDF matrix. The dimension of the TF-IDF matrix can be shown in Table 1.

Row	Column
7882	27768

Table 1. The Dimension of the TF-IDF Matrix

Question 2

After we got the data representation with TF-IDF matrix, we applied K-means clustering algorithm on the data with parameters: random_state = 0, max_iter = 1000 and n_init = 30. Comparing the result with the known class label (where we considered 'comp.sys.ibm.pc.hardware', 'comp.graphics', 'comp.sys.mac.hardware', and 'comp.os.ms-windows.misc' as one class, and considered 'rec.autos', 'rec.motorcycles', 'rec.sport.baseball', and 'rec.sport.hockey' as another class), we got the contingency table shown in Table 2.

4	3899
1718	2261

Table 2. The Contingency Table

Question 3

Besides using a contingency table, to evaluate the clustering results, we leveraged sklearn.metrics package and called homogeneity_score, completeness_score, v_measure_score, adjusted_rand_score and adjusted_mutual_info_score in sklearn.metrics to calculate the Homogeneity, Completeness, V-measure, Adjusted Rand Index and Adjusted Mutual Information Score respectively. The result can be shown in Table 3.

Measures	Value
Homogeneity	0.2535958928926043
Completeness	0.334815748824373
V-measure	0.28860033608397917
Adjusted Rand Index	0.18076179588914554
Adjusted Mutual Information Score	0.25352755133060884

Table 3. Five Measures of Clustering Results

Question 4

As we seen in previous results, K-means algorithm did not perform well on high dimensional data. It also fails to produce satisfying outcomes when clusters are not isotropically shaped or they do not have equal variances. So we need to find a better representation of our data before we perform our K-means algorithm.

First, we tried to find the effective dimension of our data by inspecting top singular values of TF-IDF matrix and deciding how many of them are significant. One guideline is to figure out the ratio of the variance of original data that is retained after dimensionality reduction. Using explained_variance_ratio_ of TruncatedSVD object, the percent of variance the top r principal components can retain v.s. r is shown in Figure 1, where r is from 1 to 1000.

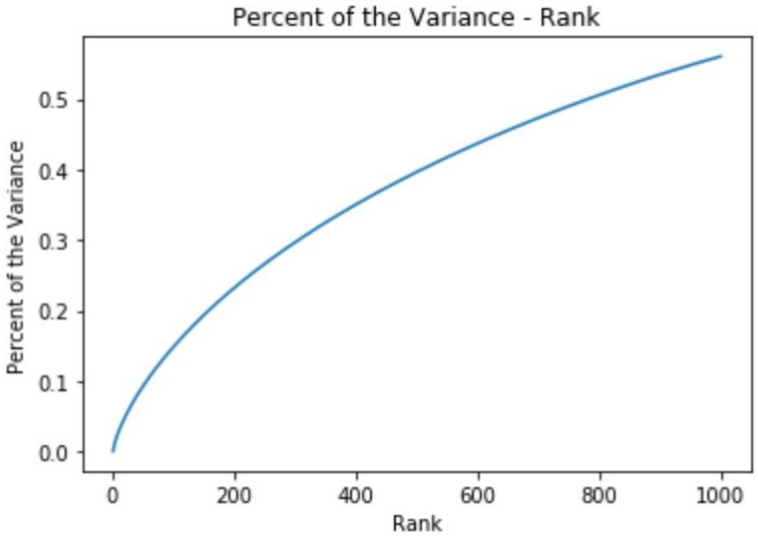

Figure 1. Percent of Variance v.s. Rank r

Question 5

Five measure scores (homogeneity, completeness, V_measure, adjusted rand index (ARI), and adjusted mutual information score (AMI)) were calculated to evaluate the quality of each clustering results on different ranks and different truncation methods, to find out the best rank of Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF) truncation methods. We performed on selected ranks as the following:

rank = [1, 2, 3, 5, 10, 20, 50, 100, 300]

And the results are as the following:

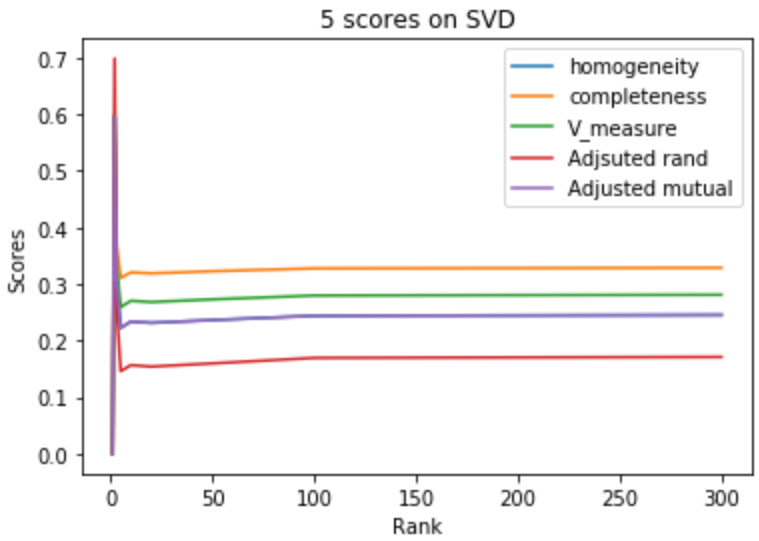

Figure 2. Five Measure Scores v.s. Rank for SVD

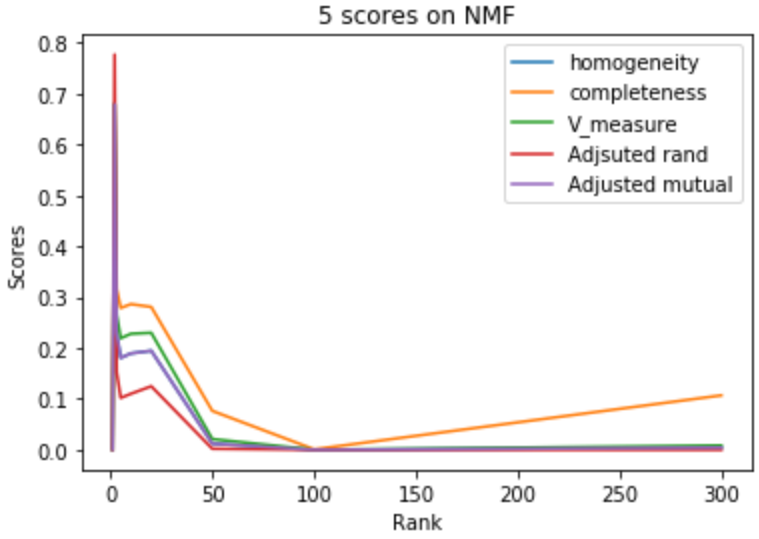

Figure 3. Five Measure Scores v.s. Rank for NMF

Since the figures above do not show clearly on rank less than 20, the zoomed figures are shown as below:

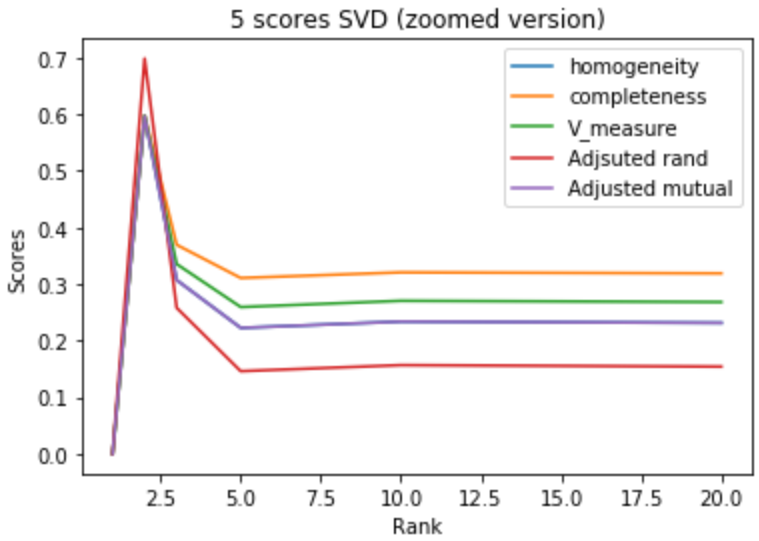

Figure 4. Zoomed Version of 5 Measure Scores v.s. Rank for SVD

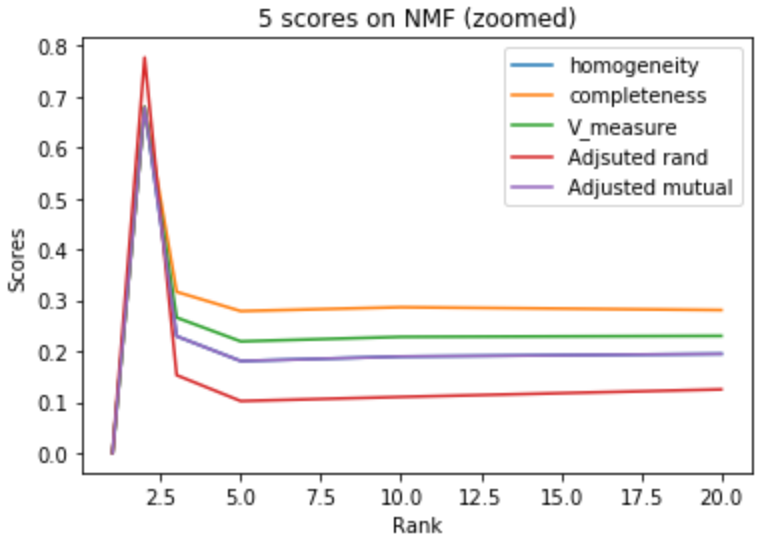

Figure 5. Zoomed Version of 5 Measure Scores v.s. Rank for NMF

From those figures, we could see that those five measure scores all achieve optimal value at rank 2 no matter on SVD or NMF truncation method. So for the following questions, we'll use rank = 2 on both SVD and NMF truncated dataset.

Question 6

In K-means algorithm, we use two norm, i.e., the Euclidean distance as the metric to represent the distance between data points. However, this is not a good metric in higher dimensions because the

distances between data points tend to be the same. Thus, as the results show that the five scores reach the best when rank is 2 and for higher rank, they are almost the same, which coincides with the theory.

Question 7

Then we directly visualized the clustered data truncated from SVD and NMF respectively, comparing to the ground truth, which the results are shown as the following:

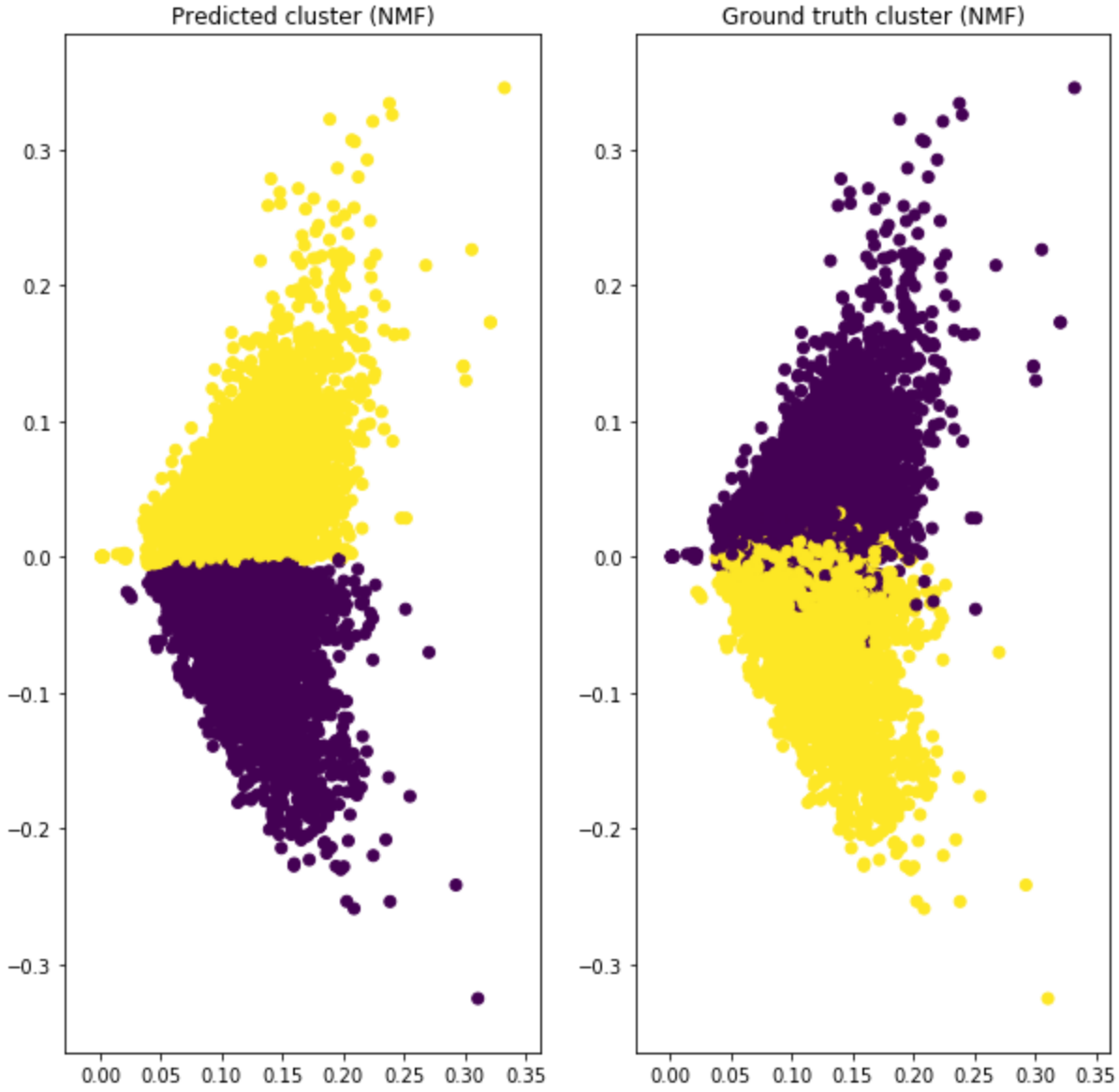

Figure 6. Predicted and Ground Truth Clustering Visualization Results for SVD

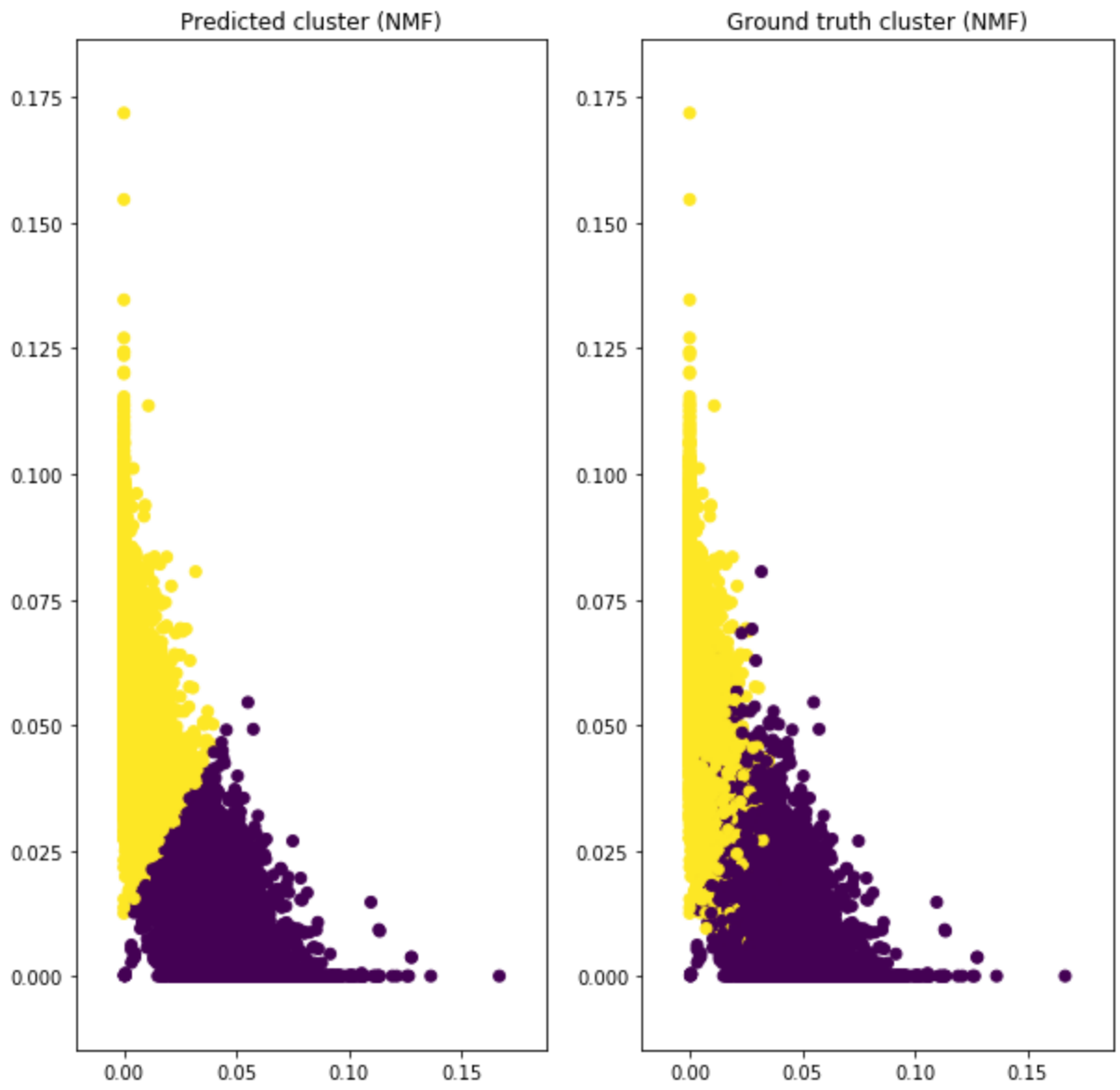

Figure 7. Predicted and Ground Truth Clustering Visualization Results for NMF

Question 8 + 10

In this part, we tried several methods to see if they could improve the clustering quality. For the data from SVD and NMF truncation, we performed scaling, logarithm, first scaling then logarithm, first logarithm then scaling transformation methods respectively to find out the best transformation method to further improve the clustering.

For scaling transformation, we used StandardScaler() method to transform the columns of the reduced-dimensional data matrix to unit variance. And for logarithm transformation, we applied the function as below:

$$f(x) = sign(x) \cdot (log(|x| + c) - log(c)) \; (c = 0.01)$$

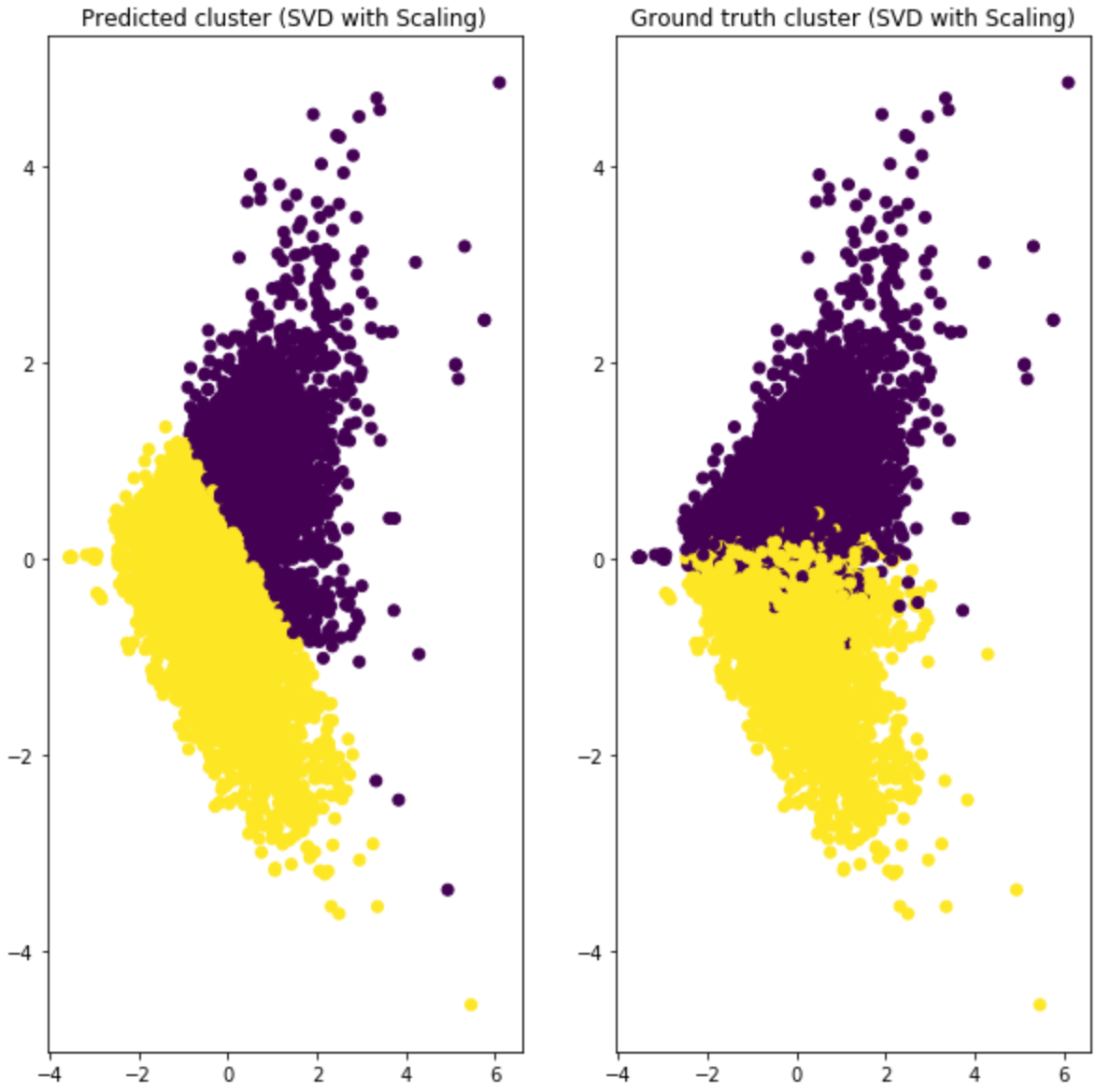

Figure 8. Visualization of Predicted and Ground Truth Clustering with Scaling Transformation on SVD Truncated Data

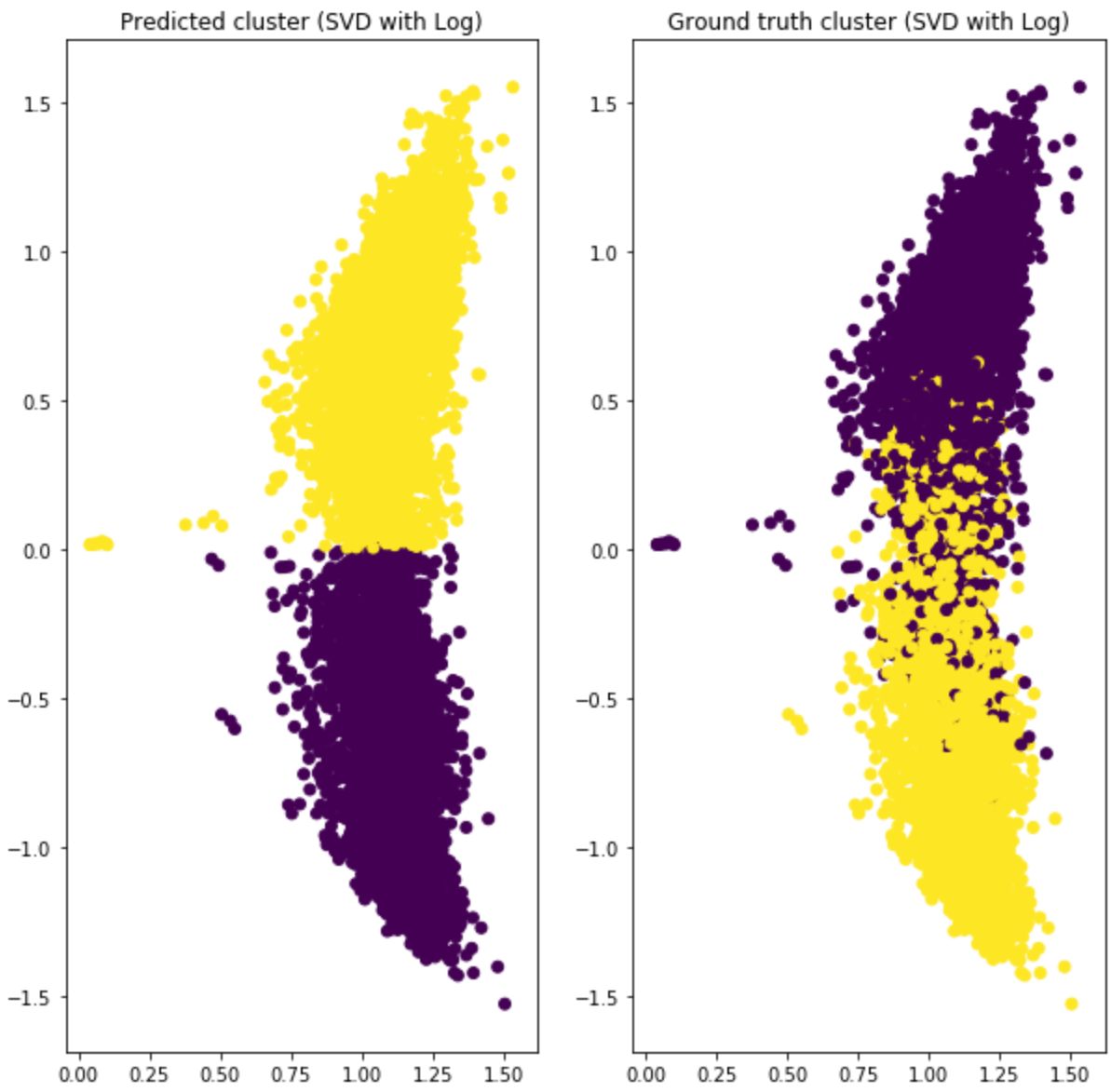

Figure 9. Visualization of Predicted and Ground Truth Clustering with Logarithm Transformation on SVD Truncated Data

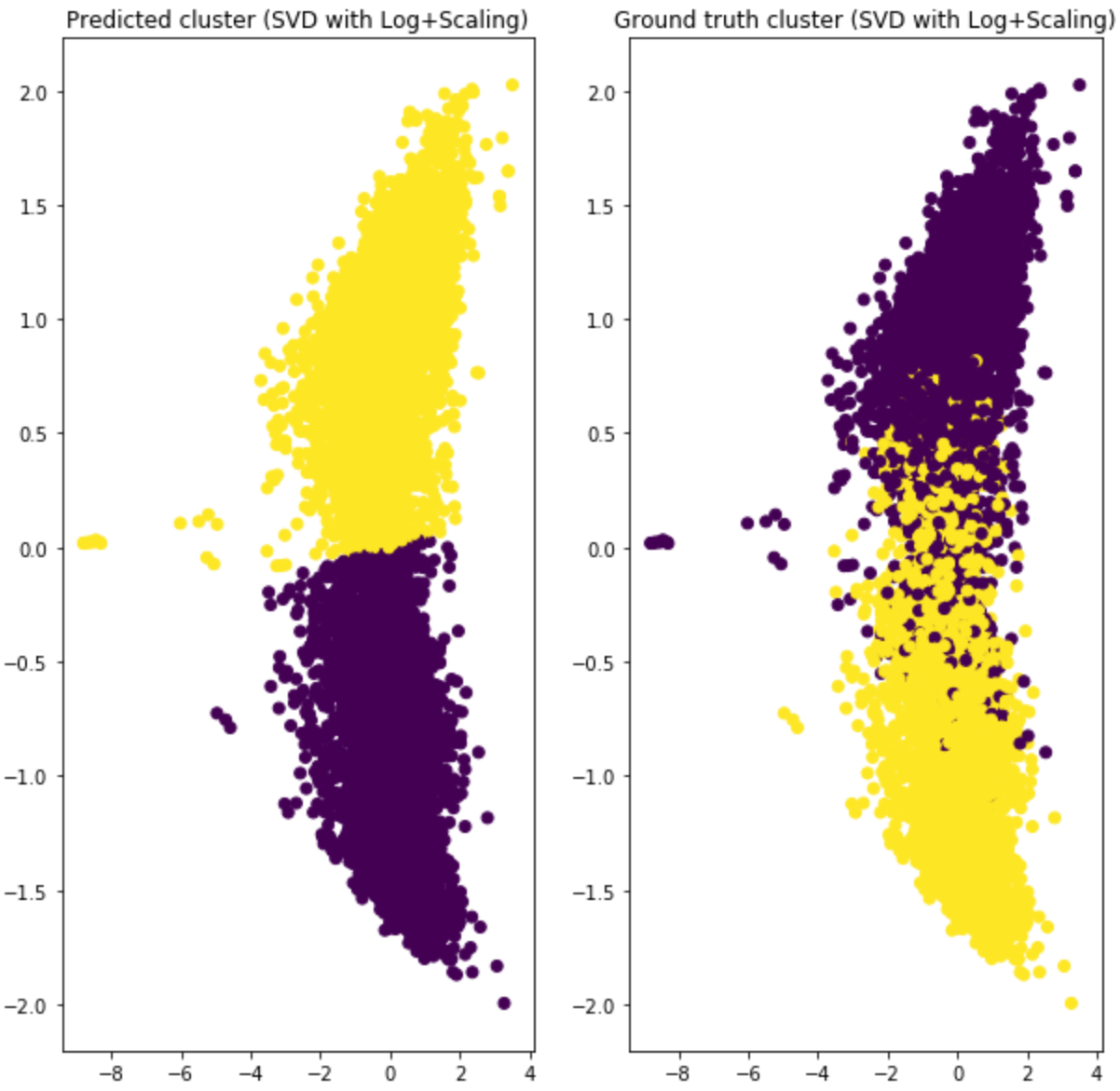

FIgure 10. Visualization of Predicted and Ground Truth Clustering with First Logarithm then Scaling Transformation on SVD Truncated data

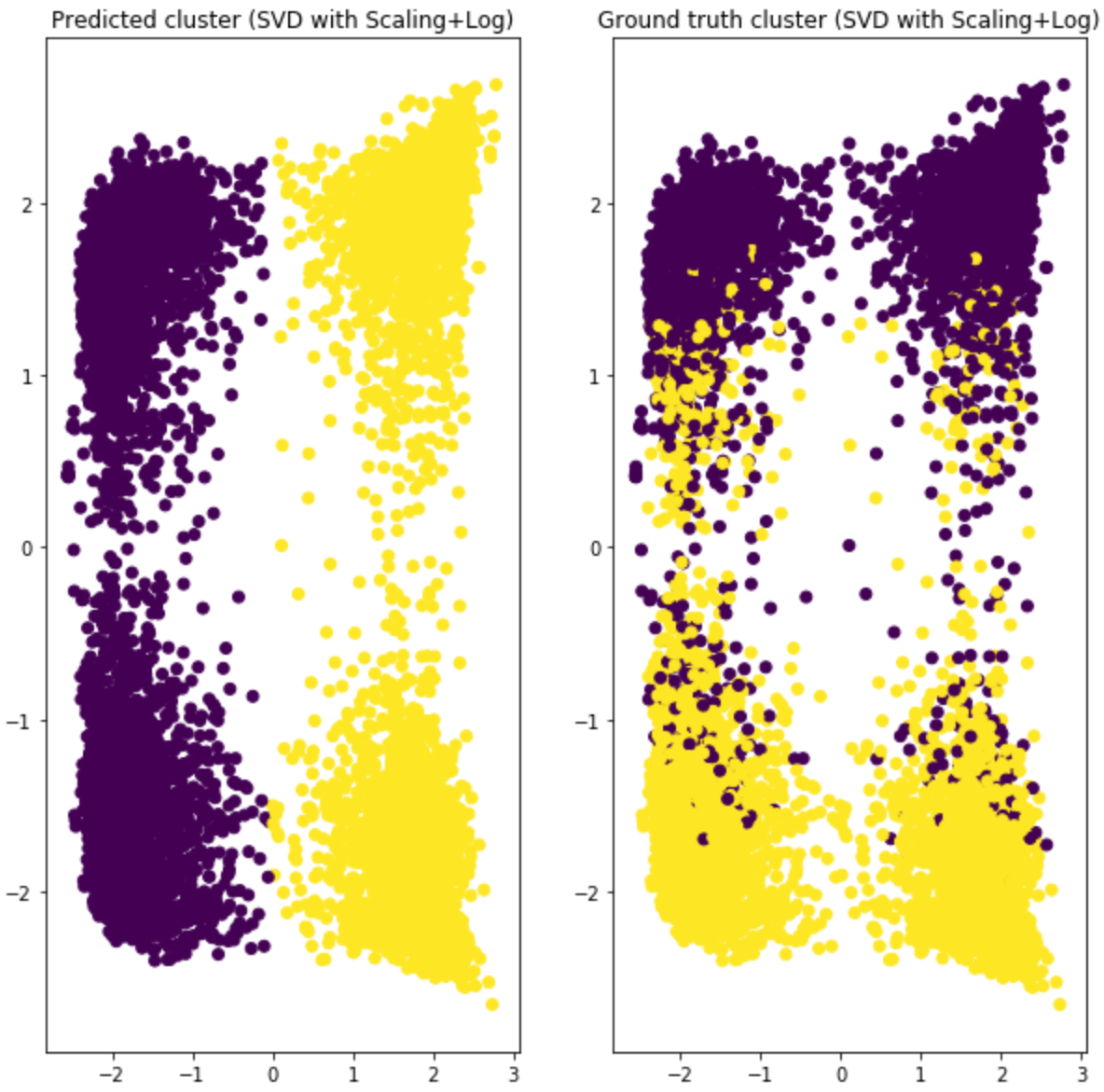

Figure 11. Visualization of Predicted and Ground Truth Clustering with First Scaling then Logarithm Transformation on SVD Truncated Data

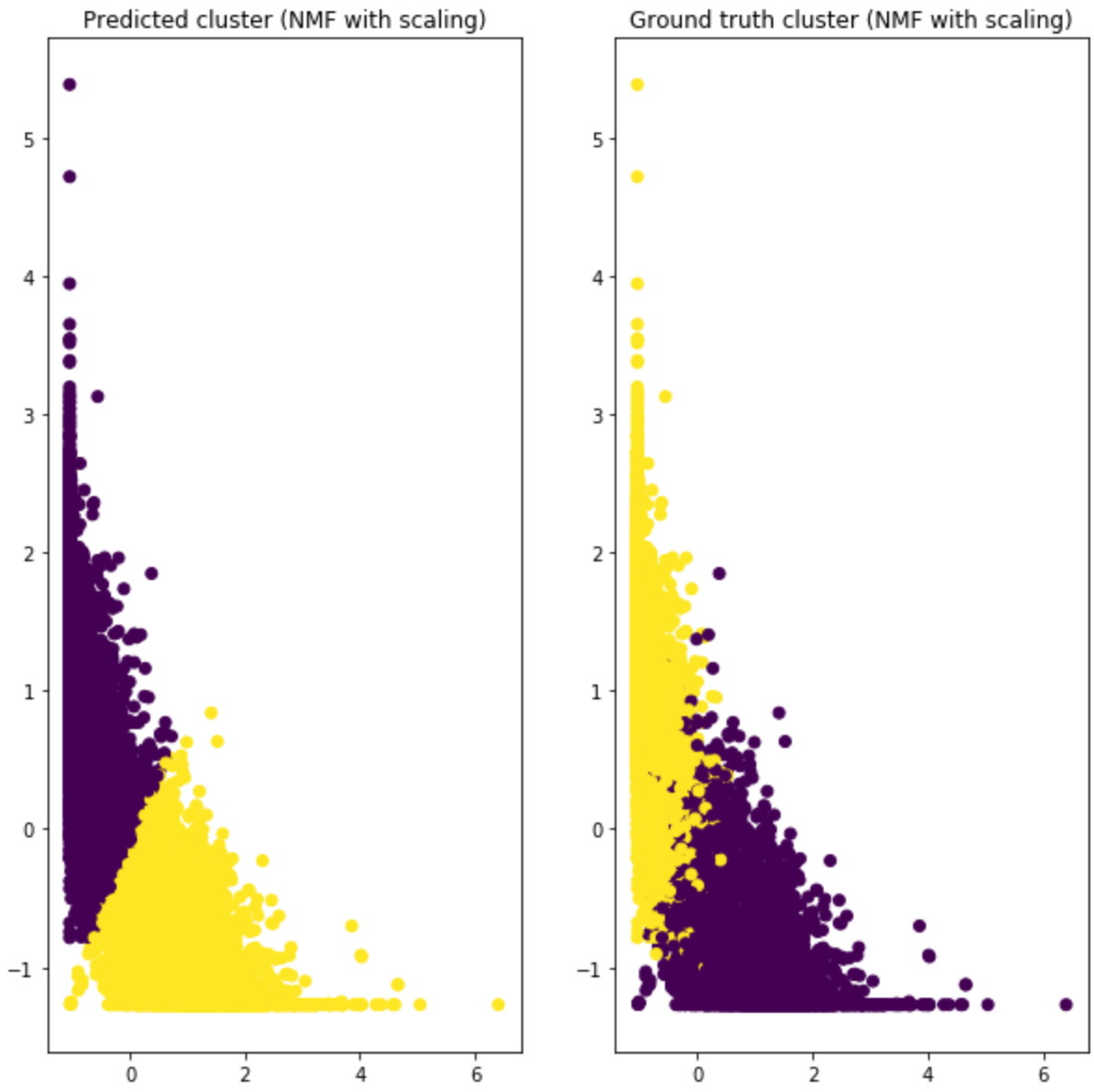

Figure 12. Visualization of Predicted and Ground Truth Clustering with Scaling Transformation on NMF Truncated Data

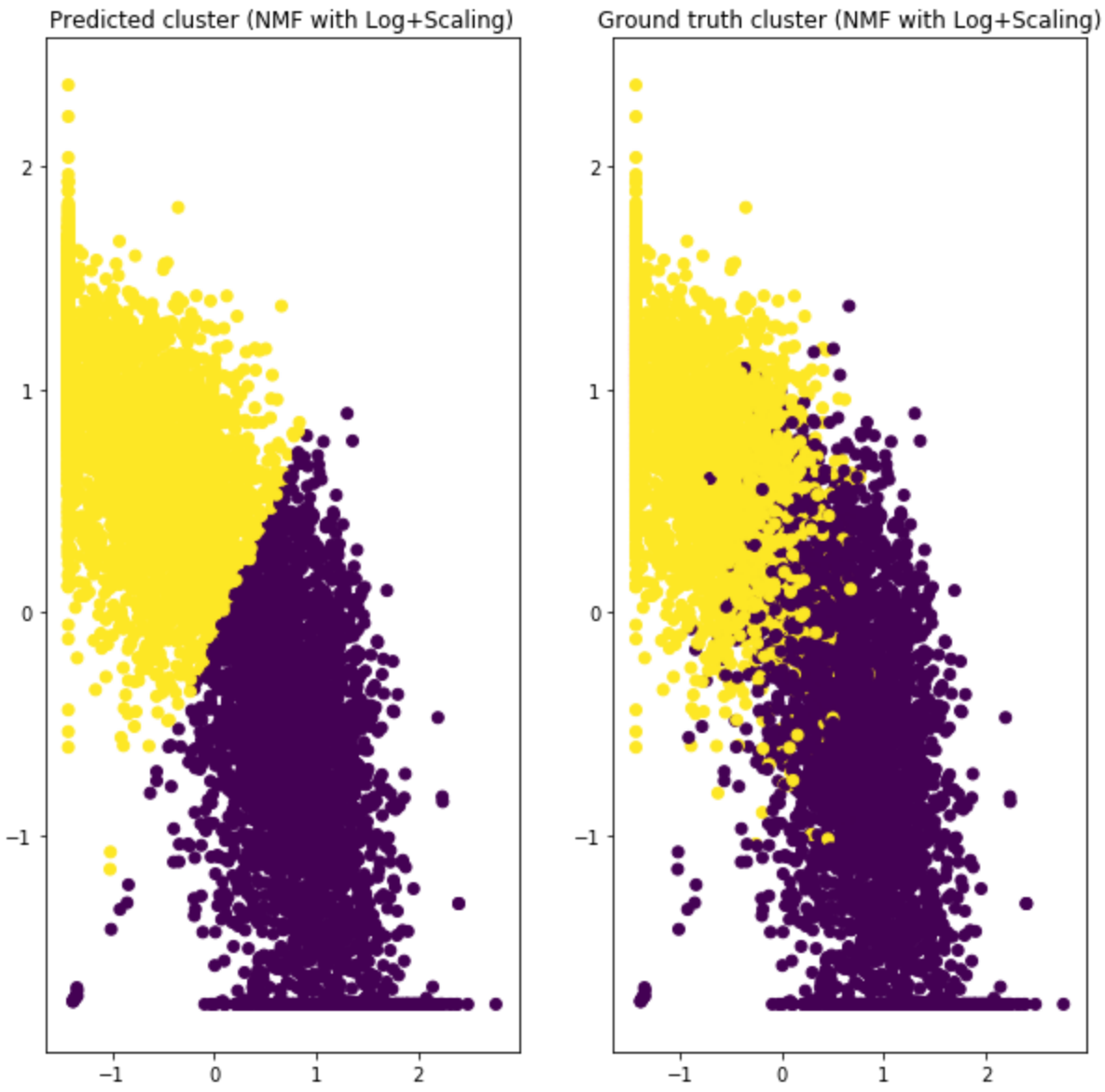

Figure 13. Visualization of Predicted and Ground Truth Clustering with Logarithm Transformation on NMF Truncated Data

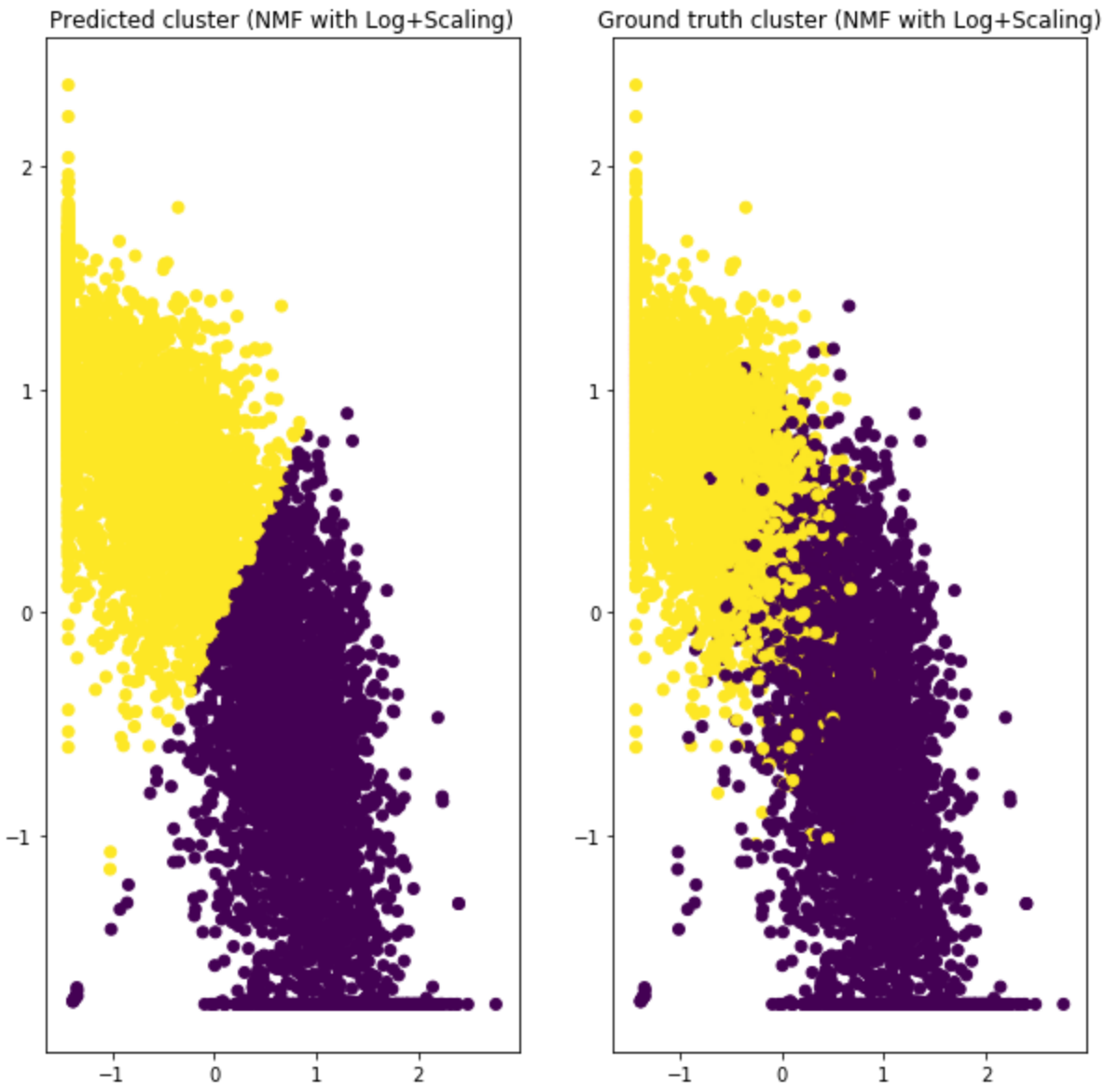

Figure 14. Visualization of Predicted and Ground Truth Clustering with First Logarithm then Scaling Transformation on NMF Truncated Data

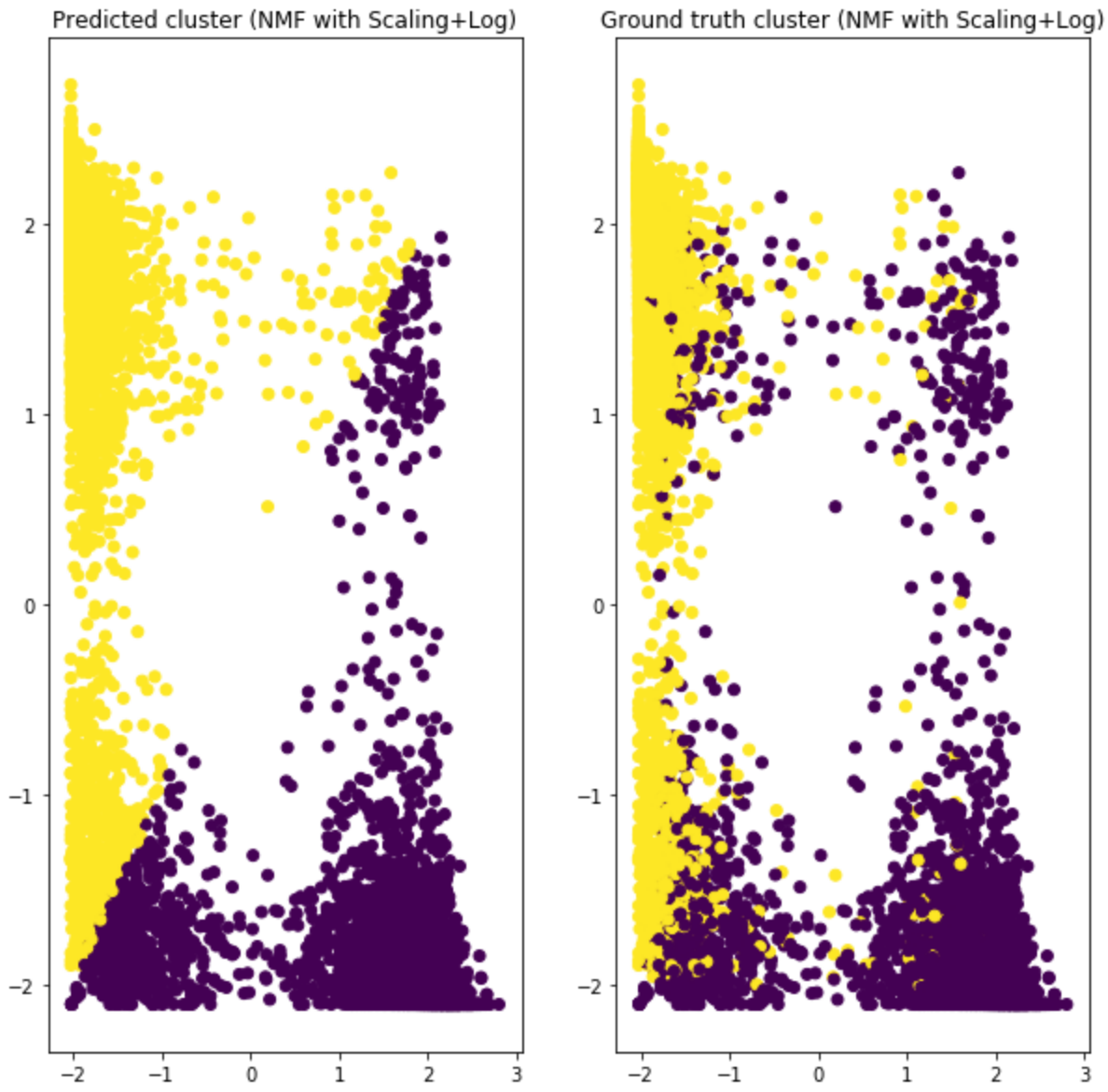

Figure 15. Visualization of Predicted and Ground Truth Clustering with First Scaling then Logarithm Transformation on NMF Truncated Data

The results of new 5 measure scores after those transformations are shown as below:

	SVD no Trans	Log Trans	Scaling Trans	Log+Scaling	Scaling+Log
Homogeneity	0.59443670811	0.61169123075	0.23512613233	0.61001785690	7.4138014e-05
Completeness	0.59559416227	0.61165941100	0.26363898333	0.60997694514	7.4298628e-05
V-measure	0.59501487231	0.61167532046	0.24856756231	0.60999740034	7.4218234e-05
ARI	0.69560459914	0.71865178779	0.25436930936	0.71693170761	-1.3196859e-05

AMI	0.59439957865	0.61162386016	0.23505610487	0.60994124082	-1.7405549e-05

Table 4. Five measure scores of different transformation from SVD truncated data

	NMF no Trans	Log Trans	Scaling Trans	Log+Scaling	Scaling+Log
Homogeneity	0.67904835623	0.67570339163	0.68280383214	0.68635124587	0.69557473047
Completeness	0.68013160921	0.67913889233	0.68564597521	0.68905514498	0.69625584032
V_measure	0.67958955105	0.67741678625	0.68422195225	0.68770053764	0.69591511874
ARI	0.77701777884	0.76498479218	0.77344267746	0.77701780324	0.79275587239
AMI	0.67901897300	0.67567370213	0.68277479272	0.68632253120	0.69554686024

Table 5. Five Measure Scores of Different Transformation from NMF Truncated Data

According to the results table, we could see that logarithm transformation works best for SVD truncated data and first scaling then logarithm transformation works best for NMF truncated data. But they are all just slightly better than the results without any transformation. The results from NMF truncated data show better score than SVD truncated data.

Question 9

The K-means algorithm uses Euclidean distance and thus implicitly makes some sort of isotropic assumption. In other words, algorithms like K-means favorably prefer spherically shaped data. The results may be really bad if one axis of data is very skewed in comparison to another. Data with heavily skewed variables may lead to very elongated clusters that are not well captured by K-means. Taking the logarithm of data will reduce the skewness and typically make the distribution more normal.

Question 11

For this question, we used the 20 categories dataset. We first transformed the data into an TF-IDF matrix just as question 1. After we got the data representation with TF-IDF matrix, we applied K-means (K = 20) clustering algorithm on the data with parameters: random_state = 0, max_iter = 1000 and n_init = 30. Comparing the result with the known class label we got the contingency matrix as table 6:

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

Table 6. The Contingency Matrix for Clustering Result of the 20 Categories Dataset

Besides using a contingency table, we also used Homogeneity, Completeness, V-measure, Adjusted Rand Index, and Adjusted Mutual Information Score to evaluate the clustering performance. The result can be shown in Table 7.

Measures	Value
Homogeneity	0.35942082651801804
Completeness	0.45111242050273204
V-measure	0.4000803165708632

Adjusted Rand Index	0.13663613501490818
Adjusted Mutual Information Score	0.35731878968094594

Table 7. Five Measures of Clustering Results of the 20 Categories Dataset

Question 12

In this part, we tried different dimensions for both truncated SVD and NMF dimensionality reduction techniques and different transformations of the obtained feature vectors as has been used above.

To find out the best number of reduced dimensions, we still performed trails on the following different dimensions:

rank = [1, 2, 3, 5, 10, 20, 50, 100, 300]

And the results are as the following:

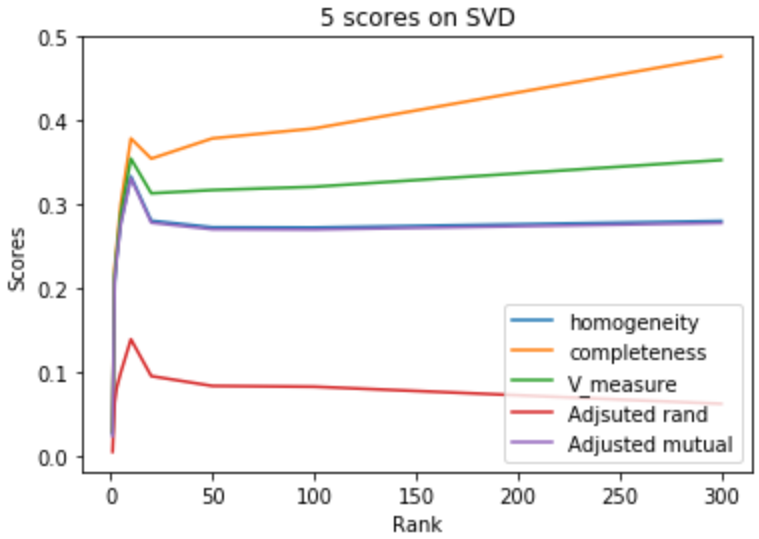

Figure 16. Five Measure Scores v.s. Rank for SVD on 20 clustering

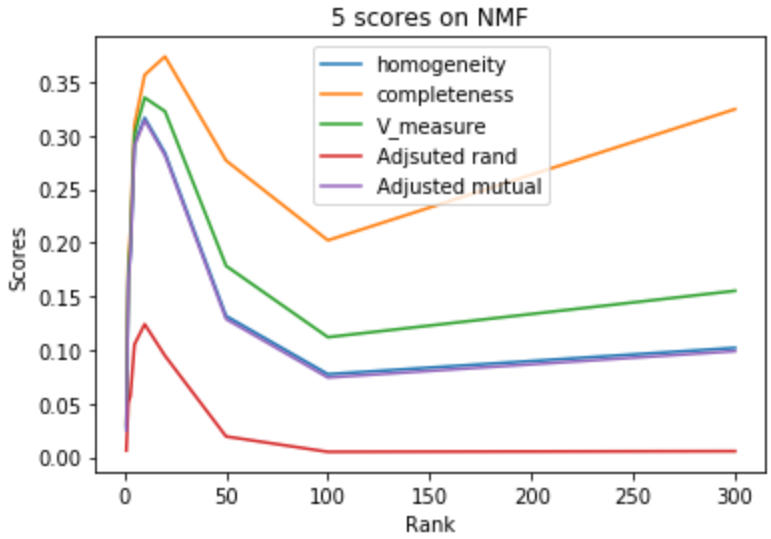

Figure 17. Five Measure Scores v.s. Rank for NMF on 20 clustering

For SVD truncation, the completeness score seems higher on higher ranks though, other scores are almost lower for higher ranks. So for the sake of comprehensive performance, we still only consider the rank less than 50 and so is the NMF truncated data. To view the results more clearly, the following pictures are the zoomed version within rank of 50.

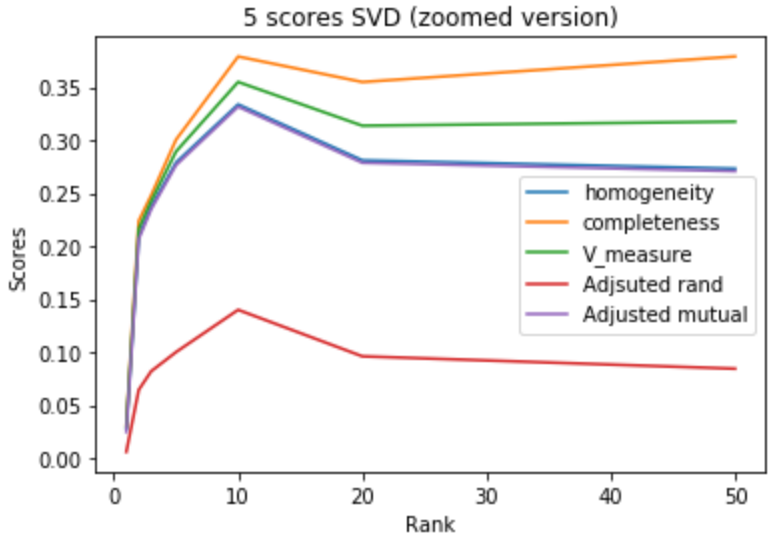

Figure 18. Zoomed version of 5 Measure Scores v.s. Rank for SVD on 20 clustering

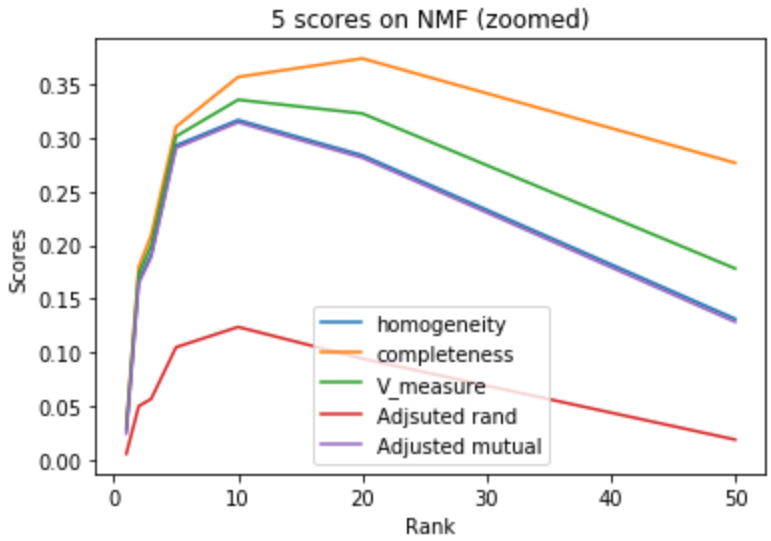

Figure 19. Zoomed version of 5 Measure Scores v.s. Rank for NMF on 20 clustering

As the results shown above, we could see that the best rank for both SVD and NMF truncated data is 10. So for the following processes, we will all use 10 reduced dimensions.

Then we performed different transformations on both truncated data, aiming to find out the best combinations according to the 5 measure scores. The tables below show the results comparing to those which have been trained directly without any transformation.

	SVD no Trans	Log Trans	Scaling Trans	Log+Scaling	Scaling+Log
Homogeneity	0.33410987331	0.23494258367	0.31499888083	0.33574126879	0.30027178767
Completeness	0.37542857014	0.24103474422	0.35538974557	0.37569764295	0.32630906840
V-measure	0.35356616167	0.23794967634	0.33397753991	0.35459742570	0.31274944440
ARI	0.13275936518	0.07906905820	0.12914175793	0.14300938193	0.11422527003
AMI	0.33194706821	0.23247181430	0.31277222274	0.33358242588	0.29800129037

Table 8. Five measure scores of different transformations from SVD truncated data on 20 clustering

	NMF no Trans	Log Trans	Scaling Trans	Log+Scaling	Scaling+Log
Homogeneity	0.31610912658	0.11534903488	0.31472831444	0.33334157529	0.30989530068
Completeness	0.35636486692	0.12983476489	0.35334206936	0.36912236595	0.34396072786
V_measure	0.33503209913	0.12216398341	0.33291927500	0.35032070322	0.32604062218

ARI	0.12359992416	0.02716062520	0.11769733242	0.12984076005	0.12753186044
AMI	0.31388615815	0.11247543270	0.31250167822	0.33117598897	0.30765057253

Table 9. Five measure scores of different transformations from NMF truncated data on 20 clustering

As we could see from the results above, first logarithm and then scaling transformation works best for both SVD and NMF truncated data. And the best transformation works only slightly better than the data without any transformation. And in all, first logarithm and then scaling transformation on SVD truncated data works slightly better than NMF truncated data with the same transformation. So the most desirable combination is SVD truncation with first logarithm then scaling transformation.

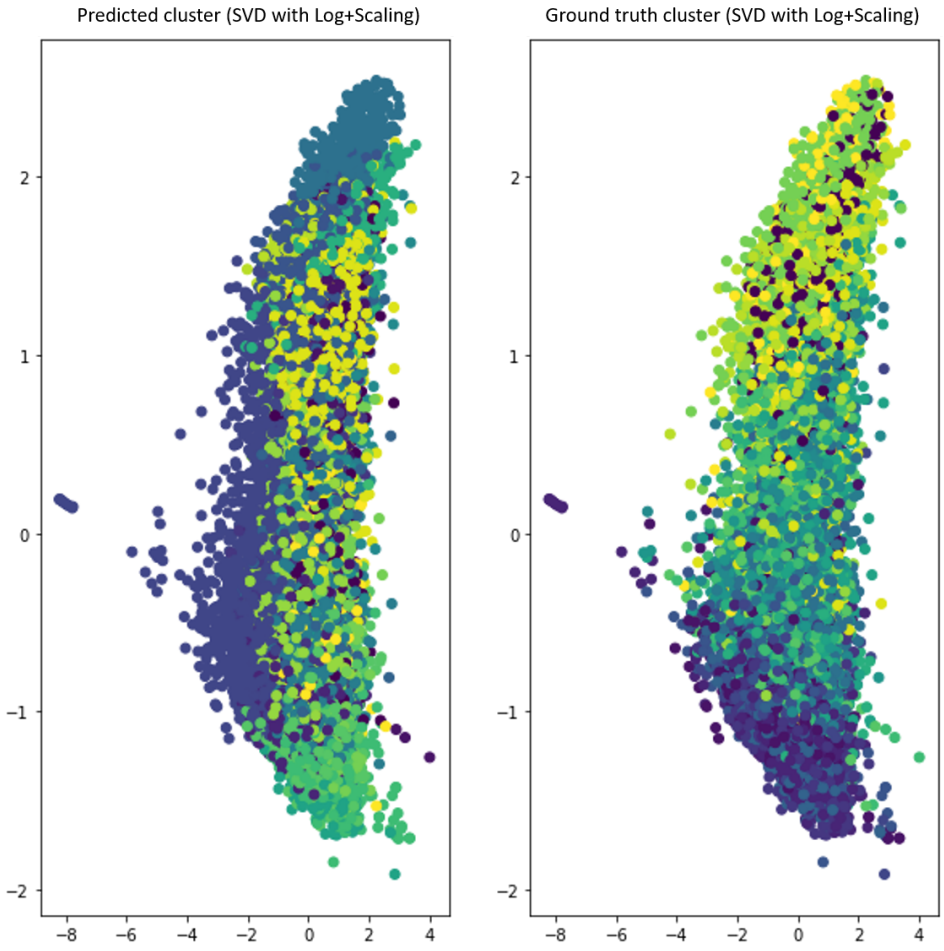

Figure 20. Visualization of Predicted and Ground Truth Clustering with First Logarithm then Scaling Transformation on SVD Truncated Data for 20 clustering