EE 219 Project 2
Clustering

Konark J S Kumar - 204759469
Shreyas Lakhe - 105026650

Dataset

In this project, we work with “20 Newsgroups” dataset. It is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups, each corresponding to a different topic. Here every topic can be viewed as a class. In order to define the clustering task, we pretend as if the class labels are not available and aim to find groupings of the documents, where documents in each group are more similar to each other than to those in other groups.
We considered the documents to be in the following classes:

Class 1	comp.graphics	comp.os.ms-windows.misc	comp.sys.ibm.pc.hardware	comp.sys.mac.hardware
Class 2	rec.autos	rec.motorcycles	rec.sport.baseball	rec.sport.hockey

Table 1: Two well separated classes

Building the TF-IDF Matrix

Here, we transformed the documents into TF-IDF vectors by setting the minimum counts of words in vocabulary to 3 and excluding the english stop words. After doing this we got a TF-IDF matrix of dimensions (7882, 27768)**.**

K-means clustering with k = 2 using the TF-IDF data

In this step we perform clustering on the documents by using the TF-IDF feature vectors that we obtained in the previous step. We obtained the following results on this task:

	Predicted Class 0	Predicted Class 1
Actual Class 0	3900	3
Actual Class 1	2296	1683

Table 2: Contingency Matrix for the clustering results

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.248	0.331	0.284	0.173	0.248

Table 3: Values of measures of purity for a given partition of the data points with respect to the ground truth.

Data Preprocessing

As we can see from the above clustering results, the performance on various purity measures is not good. This is because the TF-IDF feature which we obtained from the documents are both sparse and high dimensional.

Dimensionality Reduction

To reduce the sparse high dimensional TF-IDF features into a more compact representation to improve the clustering results, we tried out two dimensionality reduction methods, Latent Semantic Indexing and Non-negative matrix factorization.

To find the effective representation of data, and to see how many of the top singular values of TF-IDF are significant in constructing a matrix with TruncatedSVD representation we plot the variance of top r principle components can retain vs r. To do this we used explained_variance_ratio_ variable present in the TrucatedSVD object of sklearn. As can be observed from the graph below the percentage of retained variance increased with increasing number of components.

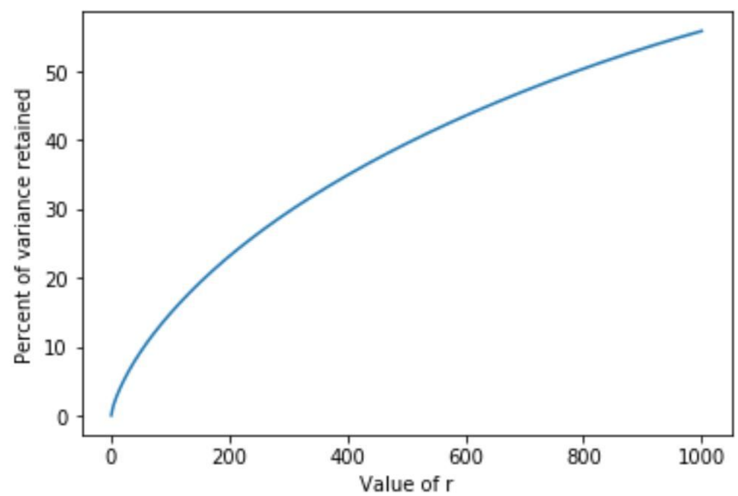


Fig 1: Percent of variance top r components can retain vs r

Using LSI

In these section, we present the results that we obtained for different values of r using Latent Semantic Indexing. As you can see from the tables below the contingency matrix is almost diagonal for around the value of r = 2 while for other values it either gives a high false positives or false negatives.

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	2202	1701
Actual Class 1	2323	1656

Table 4: Contingency Matrix for r = 1

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3699	204
Actual Class 1	446	3533

Table 5: Contingency Matrix for r = 2

	Predicted Class 0	**Predicted Class 1**

Actual Class 0	3862	41
Actual Class 1	1346	2633

Table 6: Contingency Matrix for r = 3

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	5	3898
Actual Class 1	1545	2434

Table 7: Contingency Matrix for r = 5

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3900	3
Actual Class 1	2374	1605

Table 8: Contingency Matrix for r = 10

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3	3900
Actual Class 1	1611	2368

Table 9: Contingency Matrix for r = 20

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3900	3
Actual Class 1	2352	1627

Table 10: Contingency Matrix for r = 50

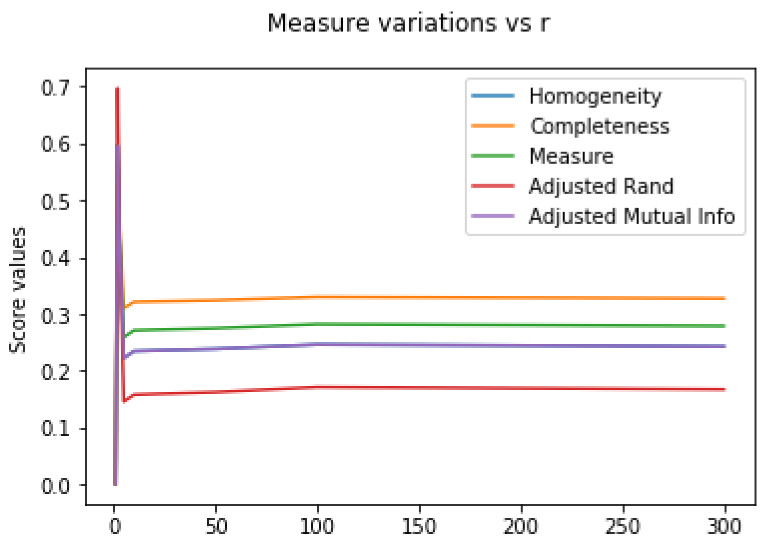
	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3900	3

Actual Class 1	2309	1670

Table 11: Contingency Matrix for r =100

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3900	3
Actual Class 1	2328	1651

Table 12: Contingency Matrix for r = 300



r Values along x-axis

Fig 2- 5 measure scores vs r for SVD

The best clustering results are obtained with r = 2 and the purity measures corresponding to those results are:

Scores	**Homogeneity**	**Completeness**	**V-Measure**	**Adjusted Rand**	**Adjusted Mutual Info**
	0.596	0.597	0.597	0.697	0.596

Table 13: Values of measures of purity for the best results

Using NMF

In these section, we present the results that we obtained for different values of r using Non negative matrix factorization. As you can see from the tables below the contingency matrix is almost diagonal for around the value of r = 2 while for other values it either gives a high false positives or false negatives.

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	2202	1701
Actual Class 1	2323	1656

Table 14: Contingency Matrix for r = 1

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3173	730
Actual Class 1	36	3943

Table 15: Contingency Matrix for r = 3

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3890	13
Actual Class 1	2305	1674

Table 16: Contingency Matrix for r = 5

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	3118	785
Actual Class 1	3977	2

Table 17: Contingency Matrix for r = 10

	Predicted Class 0	Predicted Class 1
Actual Class 0	669	3234
Actual Class 1	2	3977

Table 18: Contingency Matrix for r = 20

	Predicted Class 0	Predicted Class 1
Actual Class 0	3367	536
Actual Class 1	3977	2

Table 19: Contingency Matrix for r = 50

	Predicted Class 0	Predicted Class 1
Actual Class 0	225	3678
Actual Class 1	0	3979

Table 20: Contingency Matrix for r = 100

	Predicted Class 0	Predicted Class 1
Actual Class 0	111	3792
Actual Class 1	0	3979

Table 21: Contingency Matrix for r = 300

The best clustering results are obtained with r = 2 and the purity measures corresponding to those results are:

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.679	0.680	0.680	0.777	0.679

Table 22: Values of measures of purity for the best results

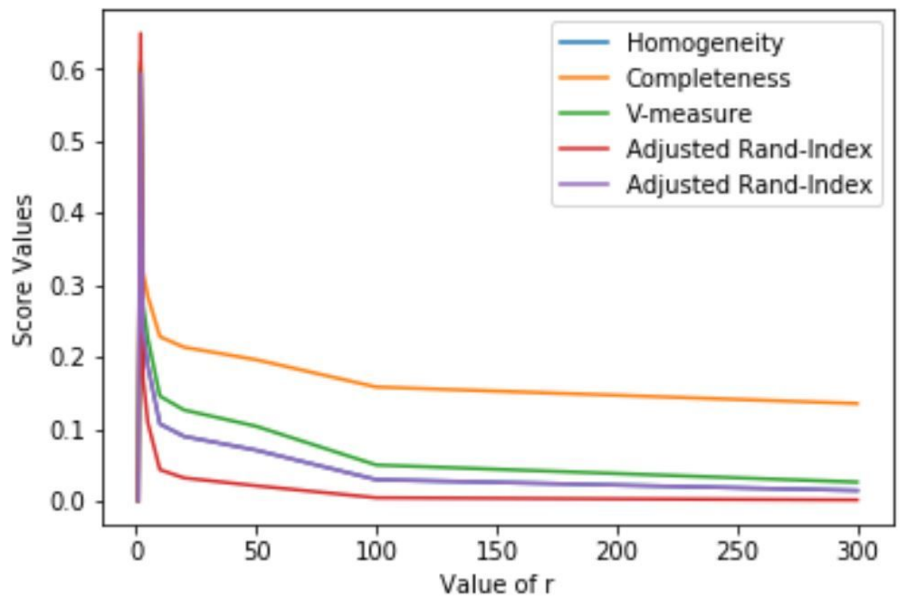


Fig 3: 5 measure scores vs r for NMF

The Non-monotonic behavior

In the graphs above regarding measure scores vs r for both NMF and SVD we observe that, it is non-monotonic and this can be credited to 'curse of dimensionality'. As the dimensionality increases the euclidean distance measures start losing their effectiveness to measure dissimilarity in highly dimensional spaces. Since K-means depend on the distance measure, it is often easier in lower-dimensional spaces where less features are used to describe the documents of interest.

Visualization of Best Case Clustering Results

Visualizing the clustering results for the best case of r = 2. The image present below the clustering results obtained on SVD reduced data.

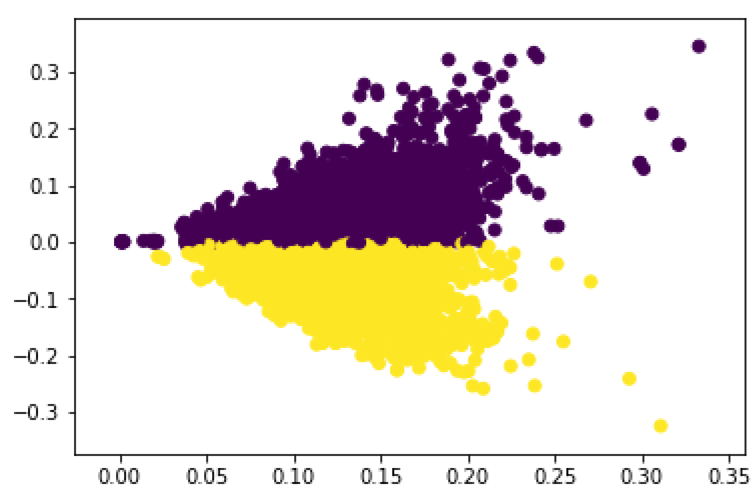


Fig 4 - Clustering results for r = 2 on SVD reduced data

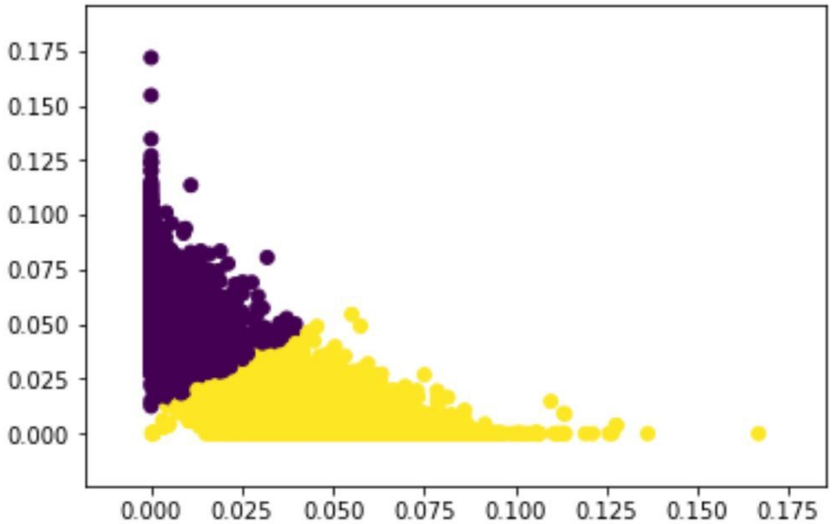


Fig 5 - Clustering results for r = 2 on NMF reduced data

Effect of various transformations on clustering Results

As we saw above, by using dimensionality reduction techniques we were able to improve the performance of the model compared to just TF-IDF. In the section, we tried experimenting with transformations on features to see if they could improve the performance of non negative matrix factorization.

Normalizing Features

As a first step we first tried normalizing the features, and as you can see from the tables below our performance improved slightly as a result of this normalization. What normalization does is convert all the features into same scale, this is important because different features might have different scales and may have varying amount of influence over the results.

	Predicted Class 0	Predicted Class 1
Actual Class 0	3462	441
Actual Class 1	80	3899

Table 23: Contingency Matrix after Normalization

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.669	0.674	0.671	0.753	0.669

Table 24 : **Purity Measures after Normalization**

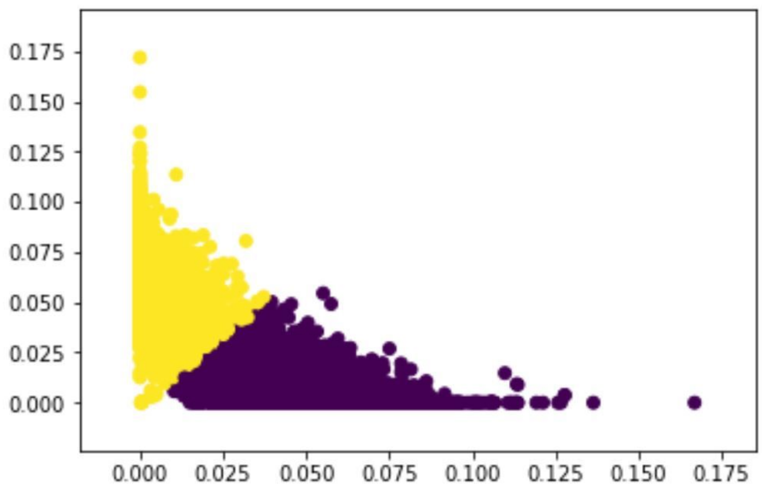


Fig 6 - Clustering results for r = 2 after Normalization

Logarithmic Transformation

In this section, we applied a non-linear transformation called logarithmic transformation which converts the skewed distribution into a more uniform one. Log transformations work only with positive data values, hence we experimented with different values of epsilon which we added to the our data values to ensure all values are positive.

	Predicted Class 0	Predicted Class 1
Actual Class 0	3681	222
Actual Class 1	176	3803

Table 25: Contingency Matrix after Logarithmic Transformation

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.712	0.712	0.712	0.808	0.712

Table 26 : **Purity Measures after Logarithmic Transformation**

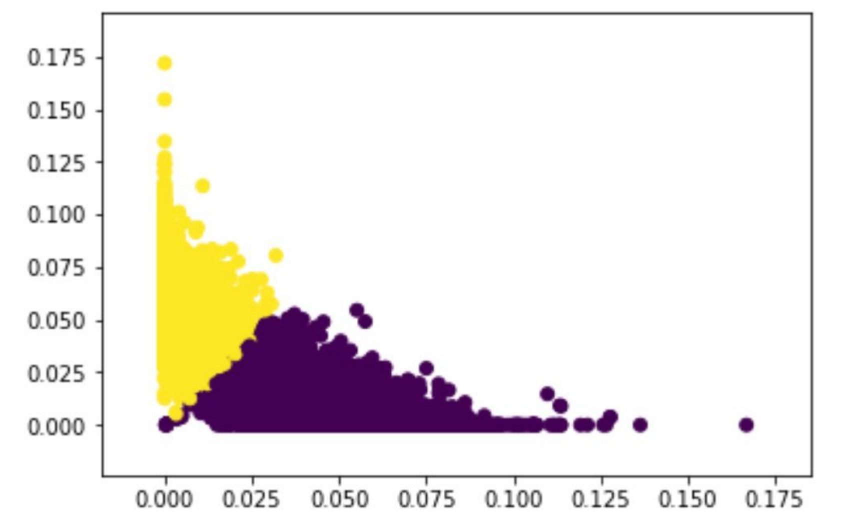


Fig 7 - Clustering results for r = 2 after Logarithmic Transformation

Normalizing and Logarithmic Transformations (and Reverse)

In this, we first normalized the features and then applied logarithmic transformations on those normalized features. In addition to this we tested with reversing the order of this transformations to see if there was any difference in the results obtained. We could see slightly better performance with first normalization and then logarithmic transformation.

	Predicted Class 0	Predicted Class 1
Actual Class 0	3602	301
Actual Class 1	123	3856

Table 27: Contingency Matrix after Normalization and Logarithmic Transformation

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.703	0.705	0.704	0.796	0.703

Table 28: **Purity Measures after Normalization and Logarithmic Transformation**

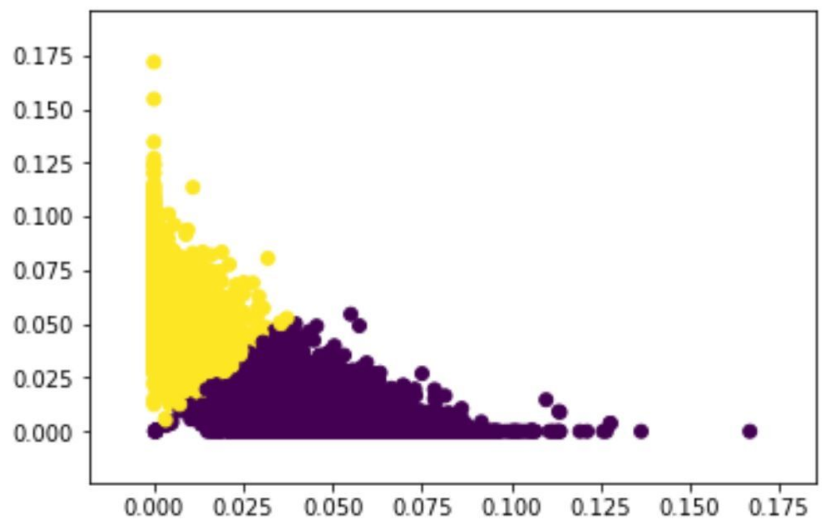


Fig 8 - Clustering results for r = 2 after Normalization and Logarithmic Transformation

Logarithmic Transformations and Normalization

	Predicted Class 0	Predicted Class 1
Actual Class 0	3522	381
Actual Class 1	105	3874

Table 29: Contingency Matrix after Logarithmic Transformation and Normalization

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.678	0.681	0.680	0.769	0.680

Table 30: **Purity Measures after Logarithmic Transformation and Normalization**

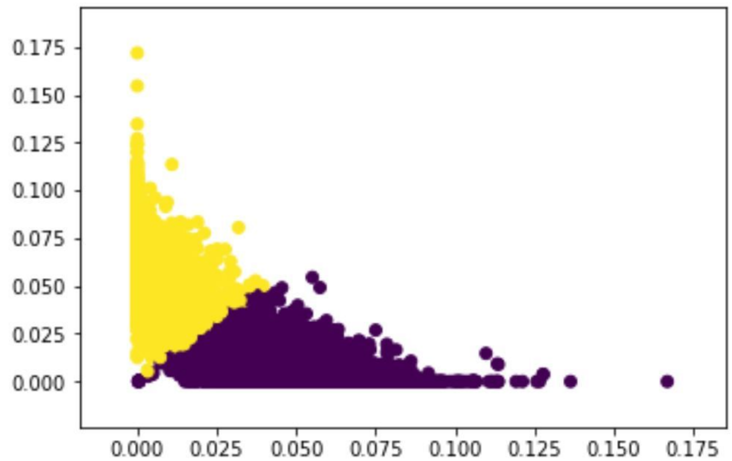


Fig 9 - Clustering results for r = 2 after Logarithmic Transformation and Normalization

CLUSTERING RESULTS FOR 20 CLUSTERS

Here, we transformed the documents into TF-IDF vectors by setting the minimum counts of words in vocabulary to 3 and excluding the english stop words. After doing this we got a TF-IDF matrix of dimensions: (18846, 52295).

Using NMF

For NMF we tested with different number of components varying from r = [1, 3, 5, 10, 20, 50, 100, 300]. The results obtained by running the k-means algorithm for 20 clusters on these components are given in the figure below. The figure contains graphs for various purity measures for these different number of components. As can be seen from the graph the best results were obtained for value of r = 10.

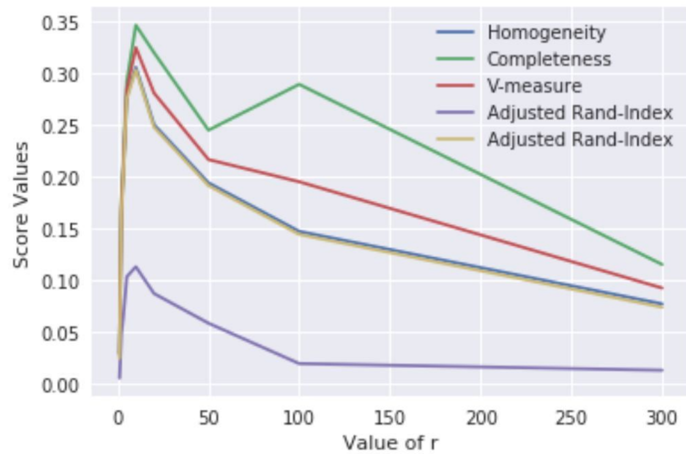


Fig 11 - 5 measure scores vs r for NMF

The purity measures that we obtained for r = 10 are given below.

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.317	0.357	0.336	0.124	0.315

Table 32 : **Purity Measures after NMF for 10 components**

The heatmap for the contingency matrix we obtained is given below

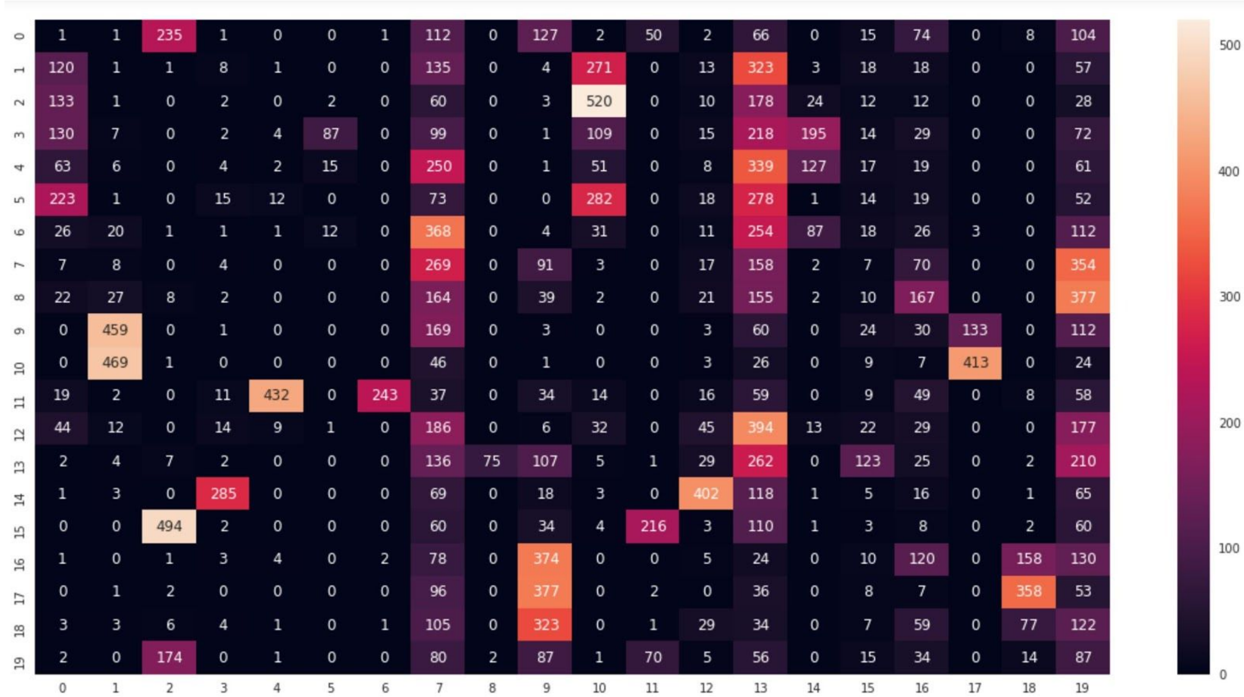


Fig 12: Heatmap showing best results obtained for contingency matrix on clustering using NMF

The clustering graph projected on 2 components using PCA is given below:

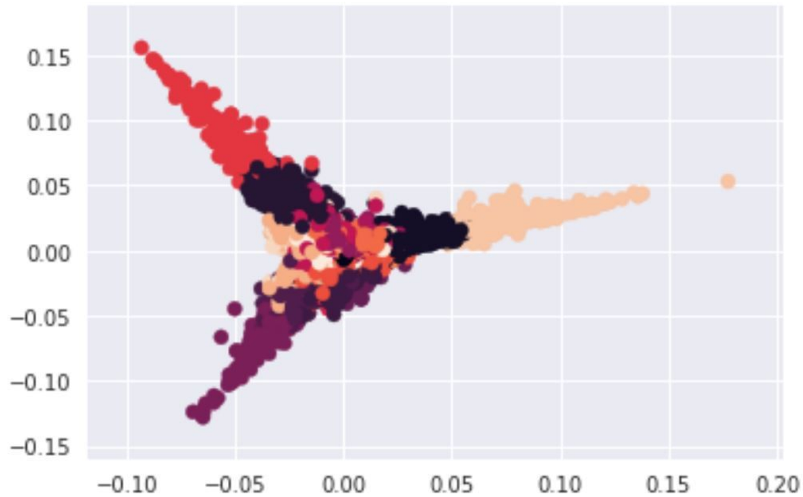


Fig 13 - Clustering results for best NMF

Normalization

After normalization of features and performing k-means clustering the best results are given below:

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.315	0.355	0.334	0.122	0.313

Table 33 : **Purity Measures after NMF for 10 components and Normalization**

The visualization of clustering results after projecting on 2 components using PCA is given below.

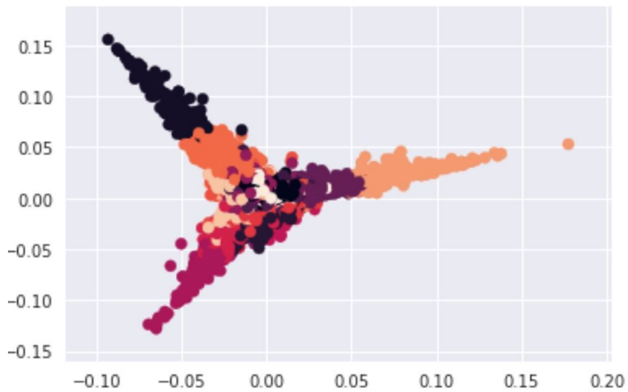


Fig 14 - Clustering results for best NMF and Normalization

The contingency matrix after normalization is given below:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	88	49	1	0	89	1	0	119	1	0	0	2	8	226	0	0	15	75	5	120
1	177	0	8	0	86	1	99	4	0	0	1	13	0	1	3	0	14	18	422	126
2	89	0	2	0	54	0	337	2	0	2	0	10	0	0	25	0	12	11	385	56
3	139	0	1	0	120	6	38	0	0	88	4	17	0	0	204	0	13	24	229	99
4	261	0	4	0	91	4	5	1	0	15	2	9	0	0	132	0	17	14	184	224
5	134	0	15	0	107	1	128	0	0	0	12	18	0	0	1	0	13	20	475	64
6	268	0	1	0	129	19	8	3	0	12	1	10	0	1	92	3	17	24	65	322
7	216	0	4	0	321	7	0	81	0	0	0	17	0	0	2	0	5	74	9	254
8	207	0	2	0	364	22	0	34	0	0	0	21	0	7	3	0	10	167	19	140
9	88	0	1	0	106	438	0	4	0	0	0	4	0	0	0	124	26	31	2	170
10	35	0	0	0	23	455	0	1	0	0	0	3	0	1	0	397	11	7	0	66
11	51	0	11	0	66	2	8	33	243	0	430	16	7	0	0	0	8	43	33	40
12	362	0	14	0	196	10	5	4	0	1	8	46	0	0	15	0	21	26	111	165
13	314	1	2	75	201	3	2	91	0	0	0	29	2	6	0	0	120	24	10	110
14	134	0	285	0	61	2	0	16	0	0	0	403	0	0	1	0	6	16	8	55
15	145	206	2	0	58	0	1	29	0	0	0	3	2	492	1	0	2	8	6	42
16	39	0	3	0	118	0	0	368	2	0	4	5	151	1	0	0	11	120	0	88
17	57	2	0	0	48	0	0	371	0	0	0	0	350	2	0	0	9	8	1	92
18	60	1	4	0	96	3	0	331	1	0	1	28	73	6	0	0	7	62	1	101
19	81	68	0	2	77	0	0	85	0	0	1	5	14	168	0	0	16	37	1	73

Fig 15: Heatmap showing best results obtained for contingency matrix on clustering using NMF and Normalization

Logarithmic Transformation

After logarithmic transformation of features and performing k-means clustering the best results are given below:

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.375	0.378	0.377	0.212	0.373

Table 34 : **Purity Measures after NMF for 10 components and Logarithmic Transformation**

The clustering results after projecting on 2 components using PCA is given below:

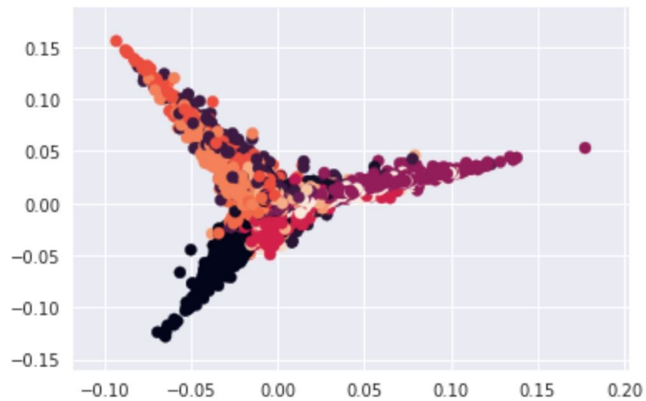


Fig 16 - Clustering results for best NMF and Logarithmic Transformation

The heatmap for contingency matrix after logarithmic transformation is given below:

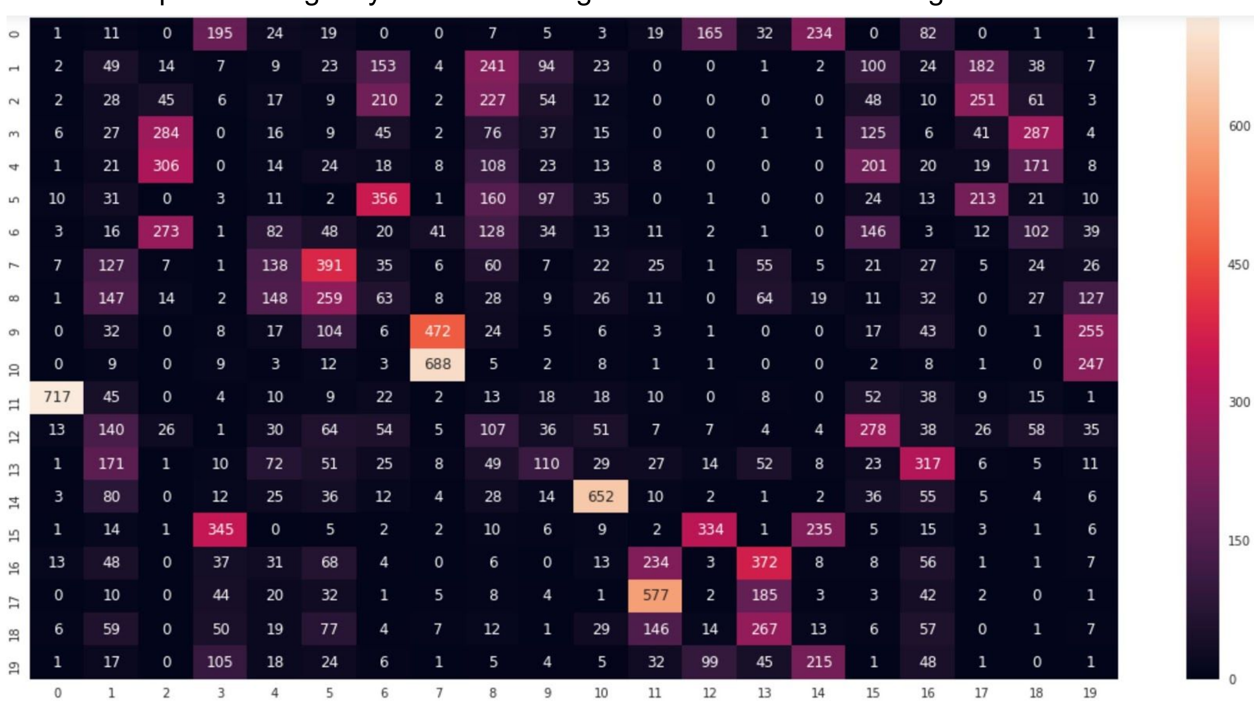


Fig 17: Heatmap showing best results obtained for contingency matrix on clustering using NMF and Normalization

Normalization and Logarithmic Transformation

After normalization and logarithmic transformation of features and performing k-means clustering the best results are given below:

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.385	0.385	0.385	0.218	0.383

Table 35: **Purity Measures after NMF for 10 components and Normalization and Logarithmic Transformation**

The heatmap for contingency matrix is given below:

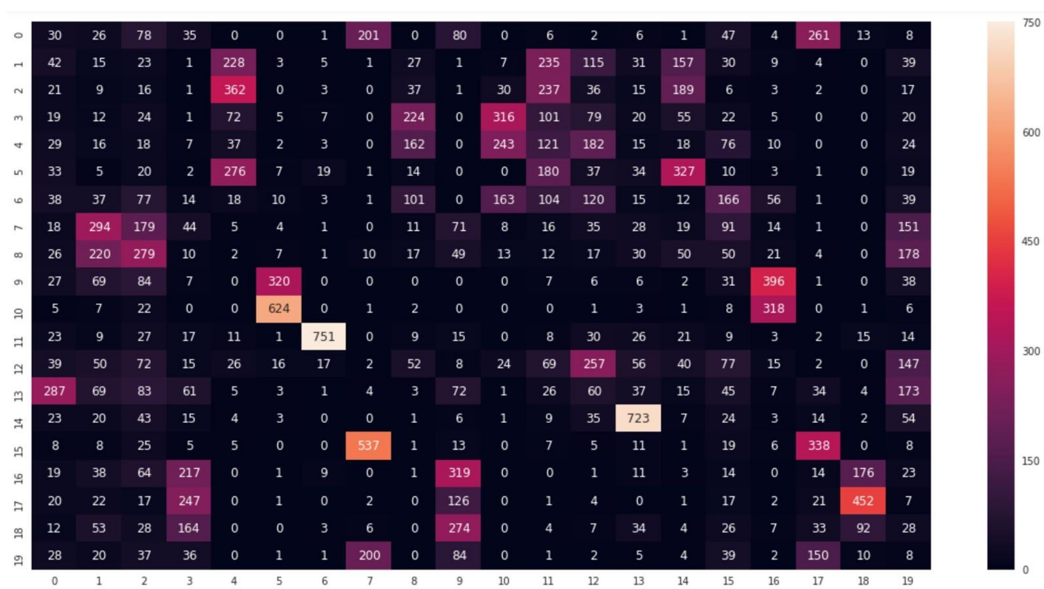


Fig 18: Heatmap showing best results obtained for contingency matrix on clustering using NMF and Normalization and Logarithmic Transformation

The clustering results after projecting on 2 components using PCA is given below:

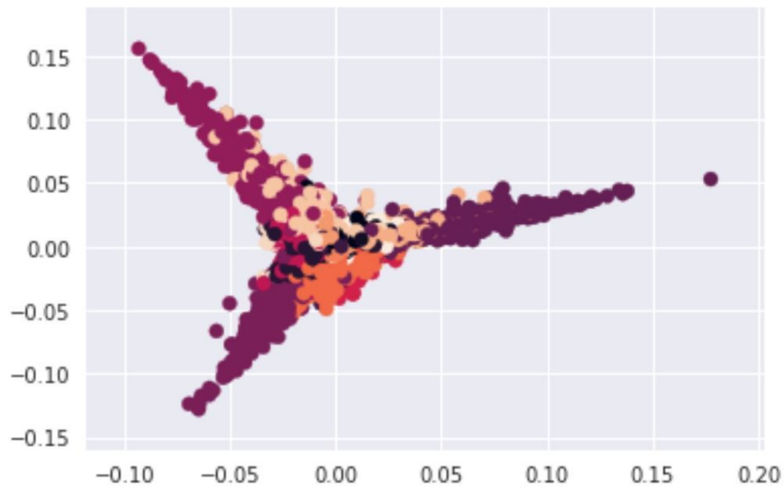


Fig 19 - Clustering results for best NMF and Normalization and Logarithmic Transformation

Logarithmic Transformation and Normalization

After logarithmic transformation and normalization of features and performing k-means clustering the best results are given below:

Scores	Homogeneity	Completeness	V-Measure	Adjusted Rand	Adjusted Mutual Info
	0.382	0.387	0.385	0.195	0.380

Table 36: **Purity Measures after NMF for 10 components and Logarithmic Transformation and Normalization**

The heatmap for contingency matrix is given below:

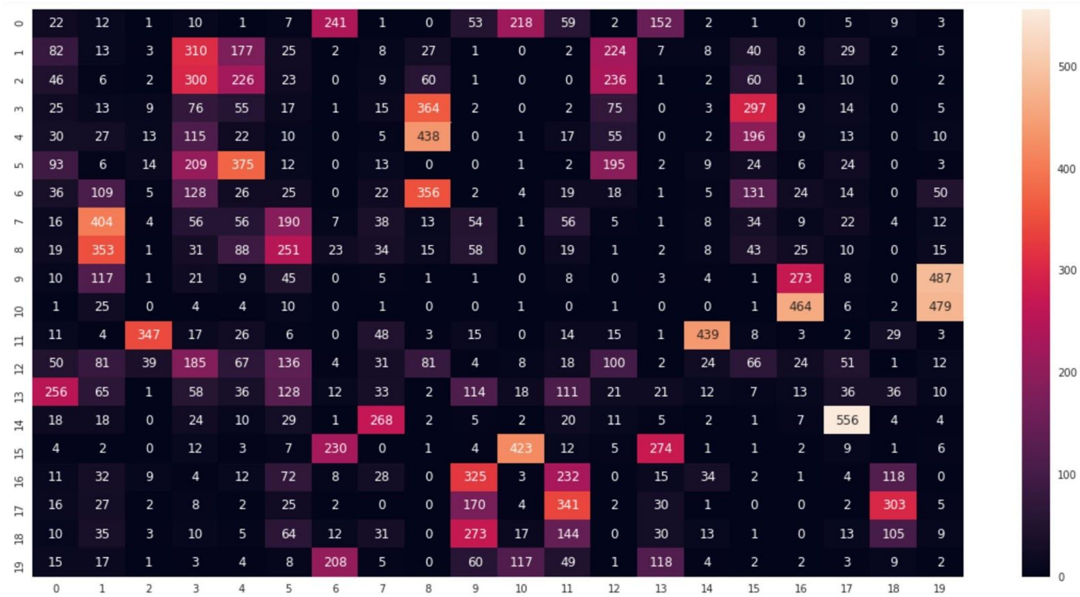
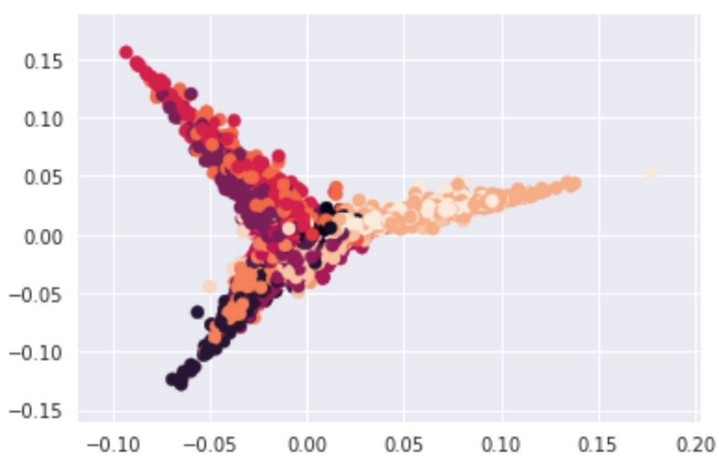


Fig 20: Heatmap showing best results obtained for contingency matrix on clustering using NMF and Logarithmic Transformation and Normalization

The clustering results after projecting on 2 components using PCA is given below:

Fig 21 - Clustering results for best NMF and Logarithmic Transformation and Normalization



Using SVD

For SVD we tested by varying the number of features/components from r = [1, 3, 5, 10, 20, 50, 100, 300]. The results obtained by running the k-means algorithm for 20 clusters on these components are given in the figure below. The figure following the table contains graphs for various purity measures vs the number of components(r). As can be seen from the graph the best results were obtained for value of r = 300.

The purity measures that we obtained for r = 300 are given below.

Scores	**Homogeneity**	**Completeness**	**V-Measure**	**Adjusted Rand**	**Adjusted Mutual Info**
	0.266	0.431	0.329	0.074	0.264

Table 37 : **Purity Measures after NMF for 300 components**

We have chosen r=300 by prioritizing the completeness score as we want to maximize the condition that all members of a given class are assigned to the same cluster.

The contingency matrix obtained on running K-means for no of components (r) equal to 300 is depicted in the heat map below.

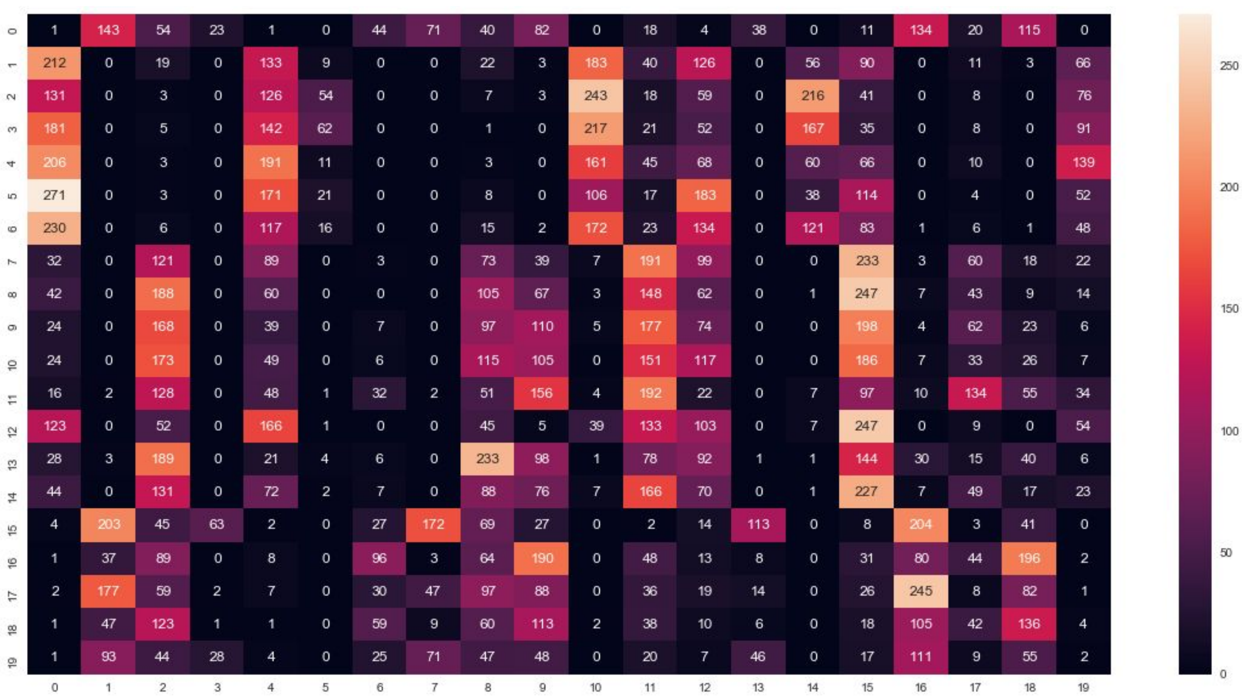


Fig 22- Contingency matrix heatmap for 20 classes and for r=300 on SVD reduced data

We then visualized the clustering results for the best case of r = 300. The image below represents the clustering results obtained on SVD reduced data.

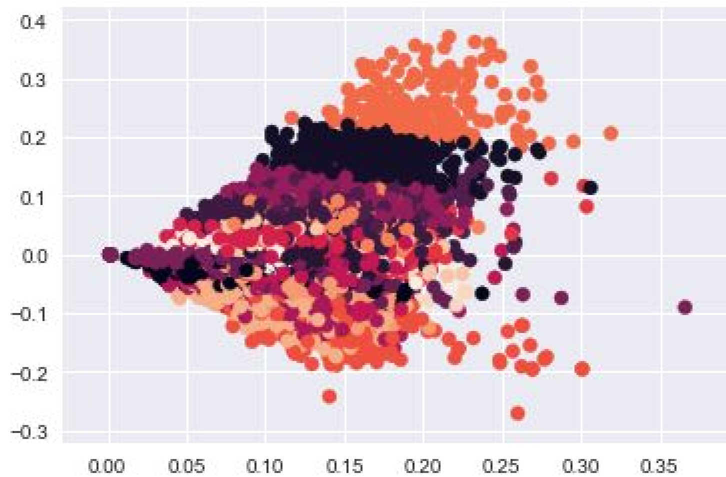


Fig 23- Clustering results for 20 classes and for r=300 on SVD reduced data