EE219 - UCLA

WINTER 2019

PROJECT 2 : REPORT

Clustering

GROUP MEMBERS

Anchal Goyanka
Nandan Parikh
Pratik Mangalore
Srishti Majumdar

INTRODUCTION

The goal of this project is to do clustering of textual data from a well defined “20 Newsgroup dataset”. The dataset deals with 20,000 documents partitioned into 20 different newsgroups like computer hardware, sport sections, religion and so on.

There are different sections to the project which can be broadly divided as follows :

- Perform K-means clustering on the dataset, and evaluate the performance of the clustering.
- Find proper representations of the data via dimensionality reduction techniques to make the clustering efficient
- Visualization of data for further inference

Part 1. Building TF-IDF matrix

In this section, we transform the input document into TF-IDF vectors using min_df = 3 and removing the stopwords.

Q 1. Dimensions

The dimensions of the TF-IDF matrix that we get is : 7882 X 27768

Part 2. Applying K-means and reporting scores

In this section, we run K-means clustering algorithm on the TF-IDF data that we get from previous question using max_iter as 1000 and n_init as 30.

Q 2. Contingency Table

Contingency table for the run is as follows:

	Class 0	Class 1
Cluster 0	4	1717
Cluster 1	3899	2262

Table 2.1: Contingency Table

Q 3. Reporting Measures

Following are the 5 measures that we get from the run:

Various Scoring Methods				
Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
0.253	0.335	0.288	0.181	0.253

Table 2.2: Scoring Measures for K-means without dimensionality reduction

Part 3. Dimensionality Reduction

For this section, we use Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF) for dimensionality reduction and note the best values for the number of components required.

Q 4. Percentage of Variance Captured: SVD

As per the question, we have 1000 principal components. The variance explained/ captured by these principal components is shown in figures 4.1 and 4.2.
In figure 4.1, we can see the the percentage of variance captured by each principal component.

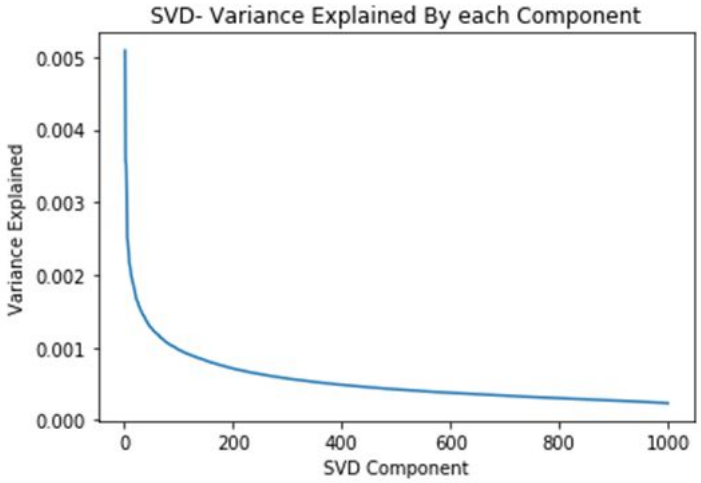

Figure 4.1: Variance captured by each principal components

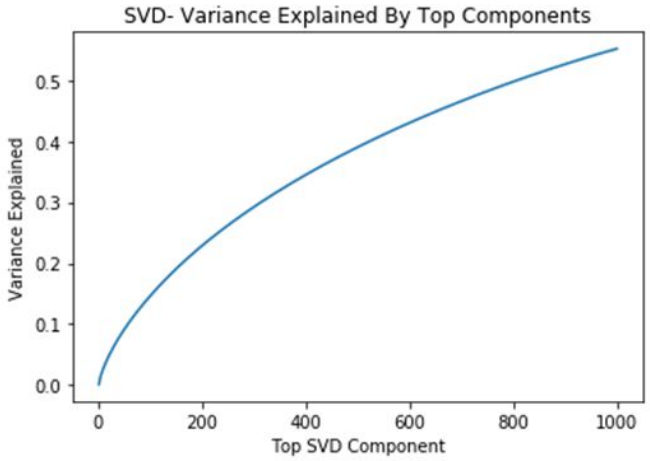

Figure 4.2: Variance captured by top r principal components

Q 5. Dimensionality Reduction Comparisons: SVD and NMF

For this question, we run SVD and NMF specified number of components and plot graphs for each method over 5 measures-homogeneity, completeness, v-measure, random index and mutual information. Figures 5.1 and 5.2 show the required plots.

SVD		**Various Scoring Methods**				
		Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
R Value	1	0.0002847	0.0002892	0.0002869	0.0003174	0.0001932
	2	0.5928125	0.5940968	0.5934540	0.6934896	0.5927752
	3	0.3998088	0.4382109	0.4181299	0.3936889	0.3997538
	5	0.2216945	0.3099640	0.2585017	0.1451558	0.2216232
	10	0.2333827	0.3202834	0.2700133	0.1563904	0.2333125
	20	0.2353359	0.3217540	0.2718423	0.1586067	0.2352659
	50	0.2418058	0.3258996	0.2776243	0.1670045	0.2417363
	100	0.2428483	0.3274116	0.2788601	0.1672120	0.2427789

	300	0.2408724	0.3259233	0.2770167	0.1649363	0.2408029

Table 5.1 : Scoring Measures : SVD

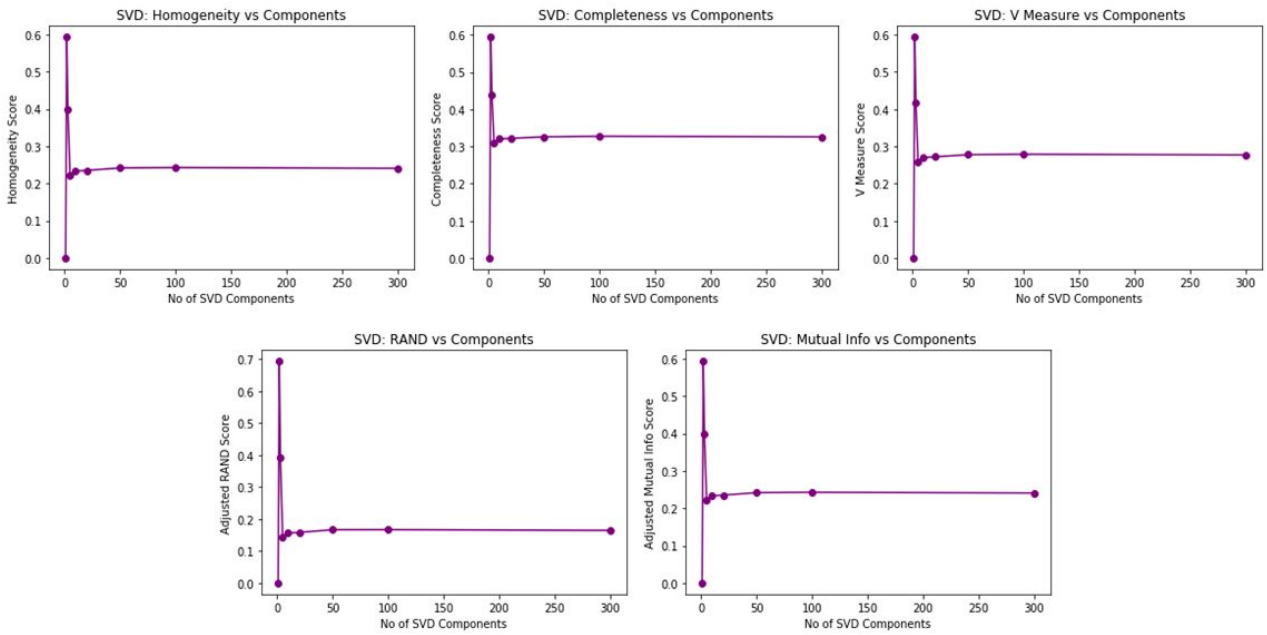

Figure 5.1: Measures of Scoring vs No of Components using SVD

NMF		Various Scoring Methods				
		Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
R Value	1	0.0002993	0.0003041	0.0003017	0.0003391	0.0002078
	2	0.6790484	0.6801316	0.6795896	0.7770178	0.6790189
	3	0.2293427	0.3164839	0.2659573	0.1527975	0.2292721
	5	0.1806307	0.2787088	0.2191989	0.1019557	0.1805557
	10	0.1890375	0.2851398	0.2273500	0.1103918	0.1889632

	20	0.1727428	0.2551113	0.2059984	0.1093814	0.1726670
	50	0.0360939	0.1455785	0.0578459	0.0076549	0.0360056
	100	0.0012877	0.0917294	0.0025398	5.520e-05	0.0011908
	300	0.0360109	0.1535455	0.0583394	0.0069995	0.0359225

Table 5.1 : Scoring Measures : NMF

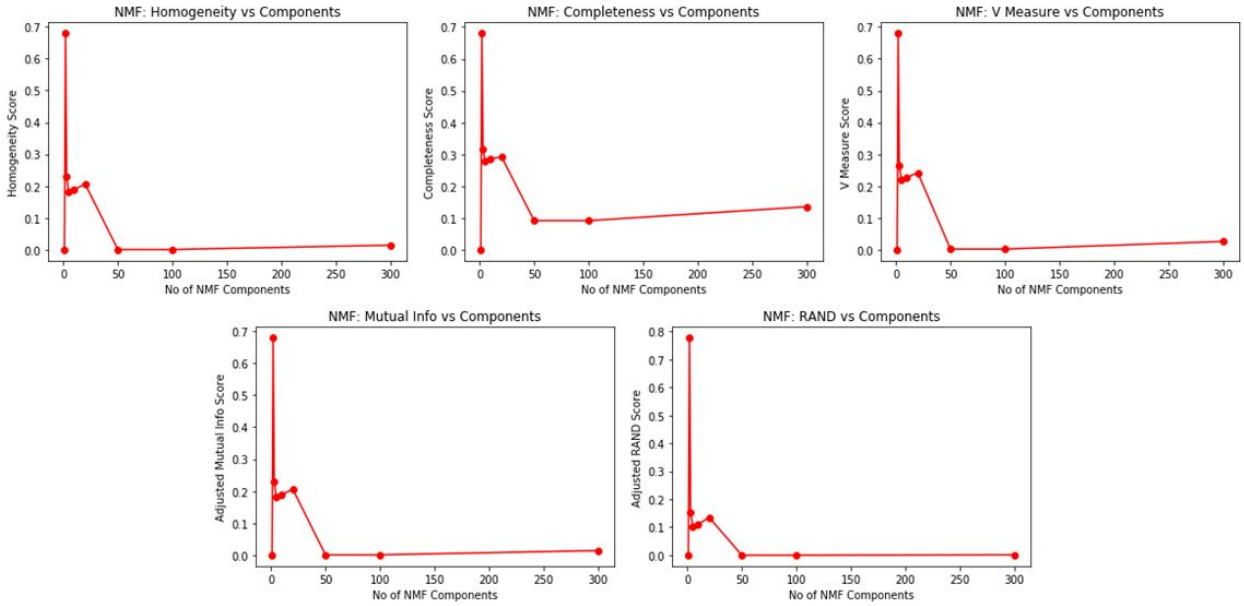

Figure 5.2: Measures of Scoring vs No of Components using NMF

As we can see from the graphs, when r=2, the scores for all five measures attain the highest value. Hence, the best choice of r is 2 for both SVD and NMF. We believe that the V-Measure is a good metric to compare the K-Means clustering results for the following reasons -

1. The V-Measure is a harmonic mean of Homogeneity and Completeness and hence somewhat includes the information from both those metrics into one.
2. The range of empirical values obtained for V-Measure were large enough to make the comparisons with more confidence. Also it was seen that if the V-Measure for a certain combination was greater than the V-Measure of another combination, then the same inequality would follow for majority of the metrics.

Q 6. Non-Monotonic Behaviour of Measures

The graph of the measures is not continuously increasing or decreasing but non-monotonic.

In general, K-means does not perform well on high-dimensional data. K-means uses euclidean distance to decide the clusters. Euclidean distance is sensitive to noise. Using euclidean distance as a distance-measure for higher dimensions, makes k-means highly sensitive to the behaviour of all dimensions. Some dimensions may conflict with each other in terms of pure l_2-distance and the clusterings thus found will be noisy.

In the case of the data given, we see k-means performs well when number of dimensions is two (there was a high peak in the graph for all the measures for both SVD and NMF at r = 2). When dimension is one, the information provided by the feature be may not enough to make good clusters. When dimensions are more than two, the dimensions/features act as noise for the euclidean measure and reduce the chances of obtaining well separated cluster.

One of the reasons for peak at r=2 could be that the data is well separated. We are using data from two different classes computers and recreation that varies highly.

Part 4. a) Visualizing the clusters :

To visualize how to data looks after clustering, we project the dim-reduced data on a 2-D plane using SVD. The color is determined by the clustering result label and x and y axis points are determined by the projection of the dim reduced data.
We plot this graph by using the best value of r, which is the best dimension according to the previous question.

Q 7. Visualizing the clustering results : SVD and NMF

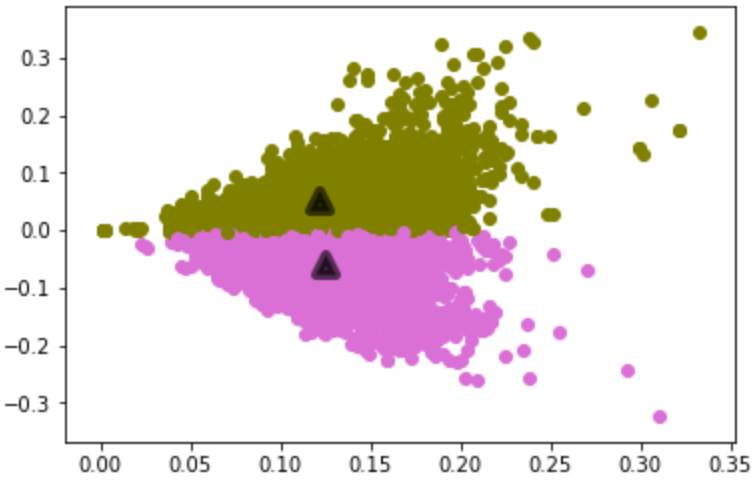

Figure 7.1: Clustering with best r of SVD with marker as cluster center

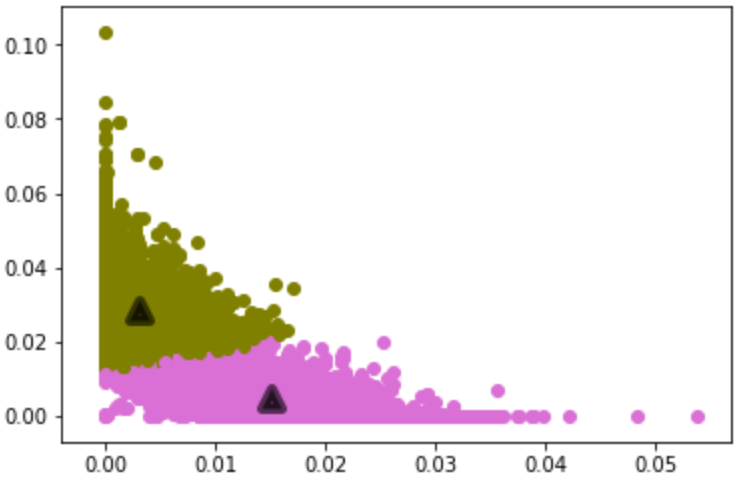

Figure 7.2 : Clustering with best r for NMF with marker as cluster center

Clustering Results		Various Scoring Methods				
		Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
Classifier	SVD	0.596	0.597	0.596	0.698	0.596
	NMF	0.679	0.680	0.680	0.777	0.679

Table 5.1 : Scoring Measures : SVD/NMF with best r

Part 4 b) Transforming and visualizing the clusters :

In this part we transform the data and then perform K-means. We perform two kinds of transformation on the data :

1. Linear - Scaling features such that each feature has unit variance
2. Non-linear transformation - Logarithm transformation

We also perform both the above transformations together in different orders. Hence we get a total of 8 visualizations as shown below.

Question 8 : Visualizing the 8 transformations as clusters in K-means

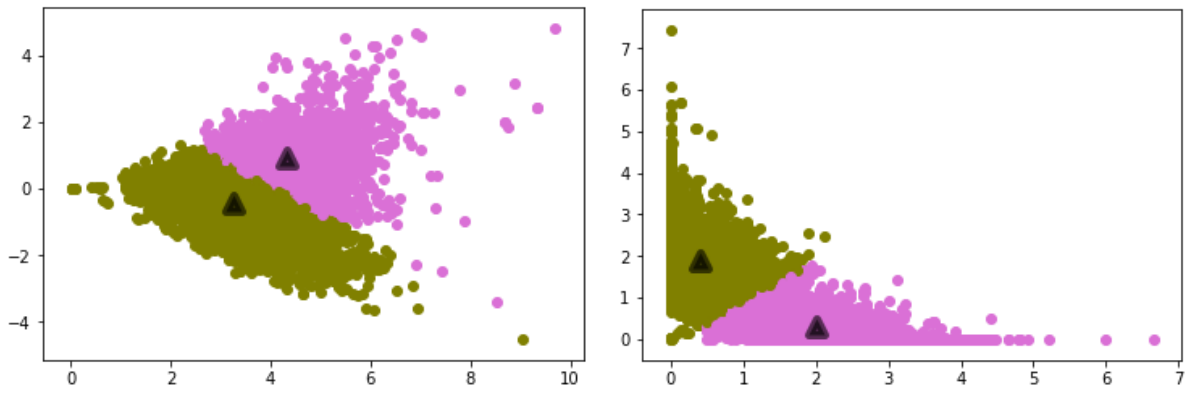

Figure 8.1 : Clustering after linear transformation (Unit Variance) - SVD vs NMF respectively

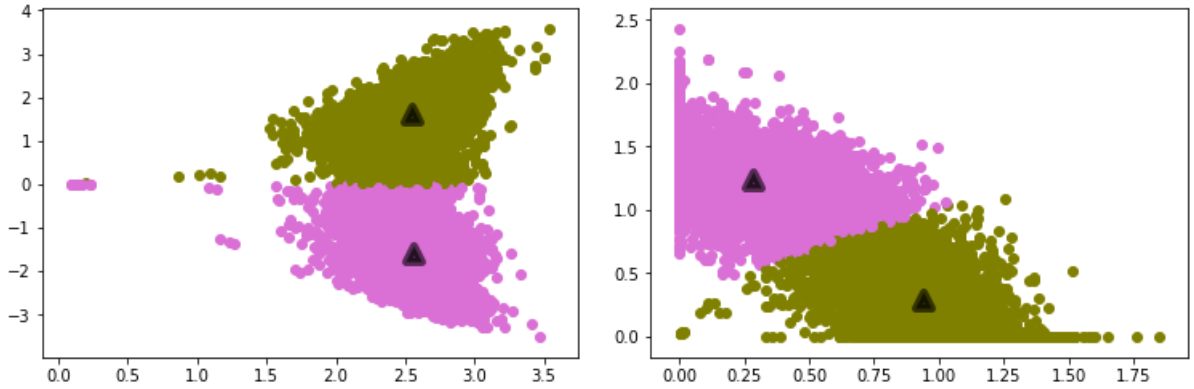

Figure 8.2 : Clustering after non-linear transformation (Log function) - SVD vs NMF respectively

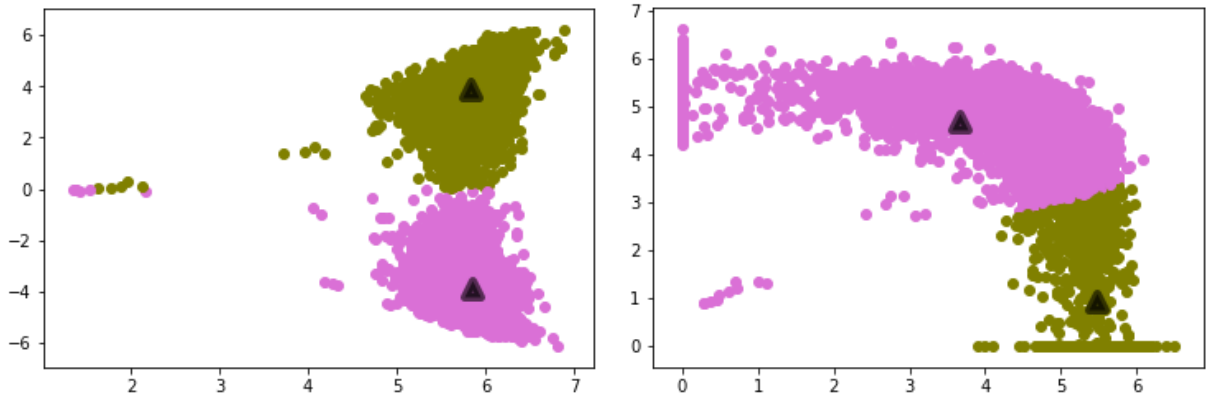

Figure 8.3 : Clustering after linear then non-linear transformation - SVD and NMF respectively

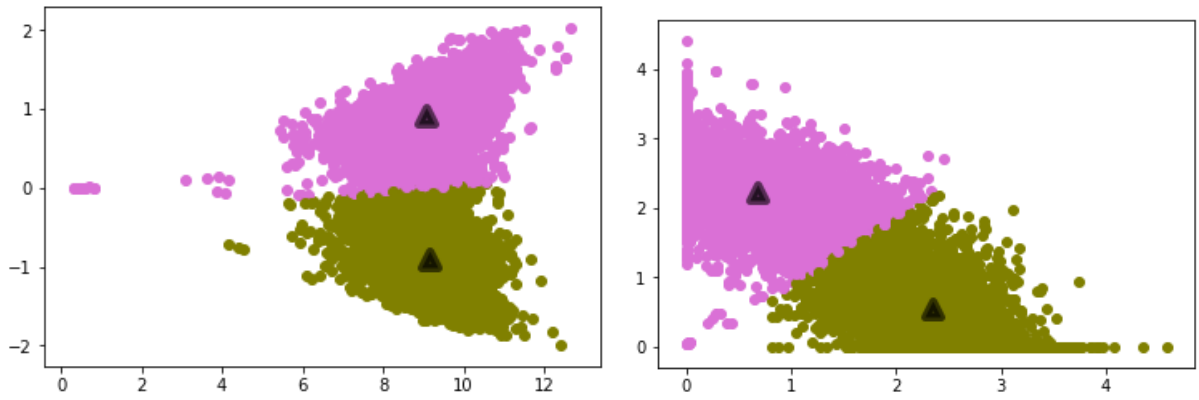

Figure 8.4 : Clustering after non-linear then linear transformation - SVD and NMF respectively

Question 9 : Justification on better results with log (non-linear) transformations

To verify how different transformation work to get clusters in K-means, we apply unit variance and logarithmic transformations on the data. There are in all 4 transformations applied on the normal SVD and NMF vectors. So we get 8 different cluster results.

Looking at the clusters and the mathematical transformations we can infer the results.
We have first applied the linear transformation of unit variance on the dimension reduced data, which normalizes the distance between different data points. We see that **SVD performs poorly in comparison to the non-transformed data as well as compared to NMF**. When we look at the graphs with normal SVD, the data already seems clustered and distributed evenly and adding this transformation probably distorts some data points.

The log transformation gives interesting insights. We see that the *data is now distributed evenly as well as lies within a smaller region* (The range has become less than half). As it can also be seen with the math, the data shrinks in a sense and is visibly close to other data points. Figure 8.2 shows how the distributed clusters look with a visible demarcation in them.
One reason why it gives better results is that the points which were very far from the normal clusters are **within** a reasonable distance from the cluster centers. The K-means cluster centers hence found are now less affected from the original distant points. The results also match with the intuition of giving better results with both SVD and NMF reductions.

The last part is using both transformations in respective orders. Applying unit variance first and the applying log on top of that gives better results for SVD but worse for NMF. For NMF, this happens as we already receive good results with variance; applying log on that distorts our result data.
When we apply log first, unit variance on top should not make much of a difference. As expected, this intuition matches with the plotted clusters.

So, the best results are found with the use of log transformation in NMF and the cluster looks particularly well divided when plotted on a 2-D dimension.

Question 10 : Clustering measures for the results

Clustering Results		Various Scoring Methods				
		Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
Classifier/ Transformation	SVD - Unit variance	0.235	0.263	0.248	0.255	0.235
	NMF - Unit variance	0.683	0.686	0.684	0.773	0.683
	SVD - Log	0.602	0.602	0.602	0.710	0.602
	NMF - Log	0.678	0.681	0.679	0.768	0.67
	SVD - Variance + Log	0.604	0.604	0.604	0.711	0.604
	NMF - Variance + Log	0.313	0.383	0.345	0.249	0.313
	SVD - Log + Variance	0.606	0.606	0.606	0.713	0.606
	NMF - Log + Variance	0.686	0.689	0.688	0.777	0.686

Table 10.1 : Clustering measures after various transformations

We mentioned the justification of these results in the previous section.

Part 5. Expanding Dataset to 20 Categories :

Question 11 : K-Means for 20 categories

The corpus was first transformed to the TF-IDF matrix as follows -

```
'''Question 11 a - TFIDF transformation'''
#tokenizer = tokenize
tfidfVectorizer = TfidfVectorizer(min_df=3, stop_words='english')
tfidf_vectors = tfidfVectorizer.fit_transform(dataset.data)

print(tfidf_vectors.shape)
```

Figure 11.1 : Dataset

The shape obtained was (18846, 52295)

K-means was performed using the following parameters (same as 2 cluster case) -

- Number of clusters = 20
- Init = ‘k-means++’
- Number of iterations = 30

Here are the results for the 5 measures -

Clustering Results	Various Scoring Methods				
	Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
TF-IDF	0.359	0.451	0.400	0.137	0.357

Table 11.1 : Clustering results with all 20 clusters

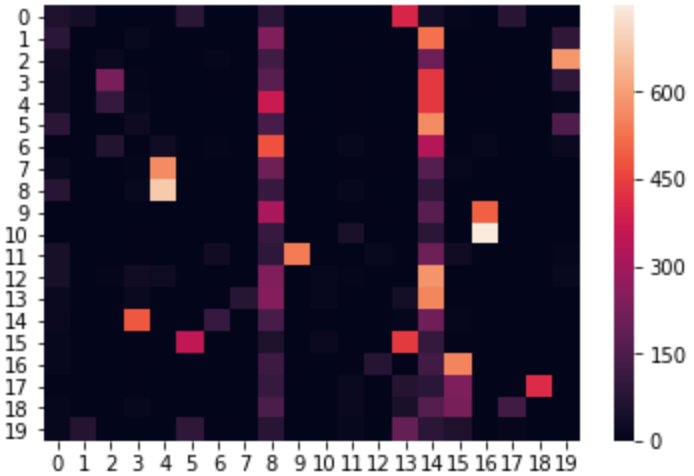

Figure 11.2 : The contingency matrix in graphic form

Question 12 : Best combination for K-Means - 20 categories

We have performed K-Means clustering for 8 combinations. Here are the results obtained for the best combination which was obtained after performing the Log Transformation on the SVD-reduced matrix with number of components r = 300.

Clustering Results	Various Scoring Methods				
	Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
SVD(r=300)	0.447	0.517	0.480	0.219	0.446

Table 12.1 : Best clustering Results

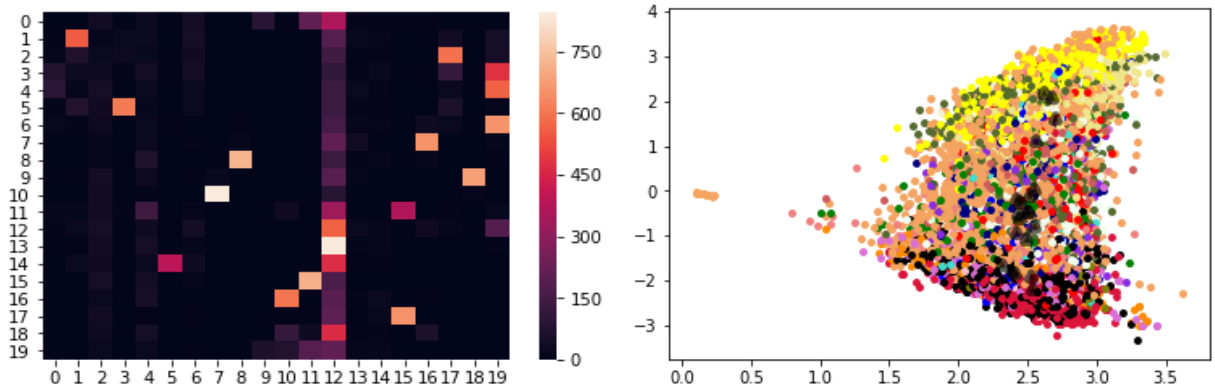

Figure 12.1 : On the left is the contingency matrix obtained for the same, and on the right is the clustering result.

The combination which performed the second best was SVD with Unit Variance followed by log transform. We obtained a score of 0.457 for the V-Measure and hence the difference between scores is 0.48 - 0.457 = 0.023.

Below is a matrix which shows the difference with the best combination for each combination we have used. Note:- r = 300 for every SVD combination and r = 10 for every NMF combination since r = 10 was obtained as the best r, based of the logic mentioned below. Each row corresponds to a combination. Each column represents a metric. The value present in every cell in the matrix below represents the amount by which the score for SVD with Log Transform was better than the score for the combination given by the row. Since every value we see is mostly positive, it means SVD with Log Transform in general performed better than the other combinations.

Quantitative Difference	Various Scoring Methods				
	Homogeneity	Completeness	V-measure	Adjusted Rand	Adjusted Mutual Info
SVD + Unit Variance	0.305	0.272	0.301	0.181	0.307
NMF + Unit Variance	0.132	0.164	0.147	0.101	0.134
SVD + Log Transform	0	0	0	0	0
NMF + Log Transform	0.072	0.13	0.099	0.015	0.073
SVD + Unit Variance -> Log Transform	0.011	0.038	0.023	-0.004	0.011
NMF + Unit Variance -> Log Transform	0.138	0.203	0.168	0.068	0.139
SVD + Log Transform -> Unit Variance	0.14	0.087	0.122	0.108	0.142
NMF + Log Transform -> Unit Variance	0.077	0.14	0.017	0.02	0.078

As seen from the above matrix, it is generally useful to take the Log Transform of the reduced component matrix as it generally tends to smoothen the influence of outlier points (points far away from the class centroids).

The reason this combination was selected as the best not only because it has the highest score for the V-Measure. The value of r = 300 for SVD was selected not only because it has the highest score for the V-Measure, which we have decided to prioritize as a metric, but also because it has the highest score for majority of the metrics K-Means was fit for, as seen below -

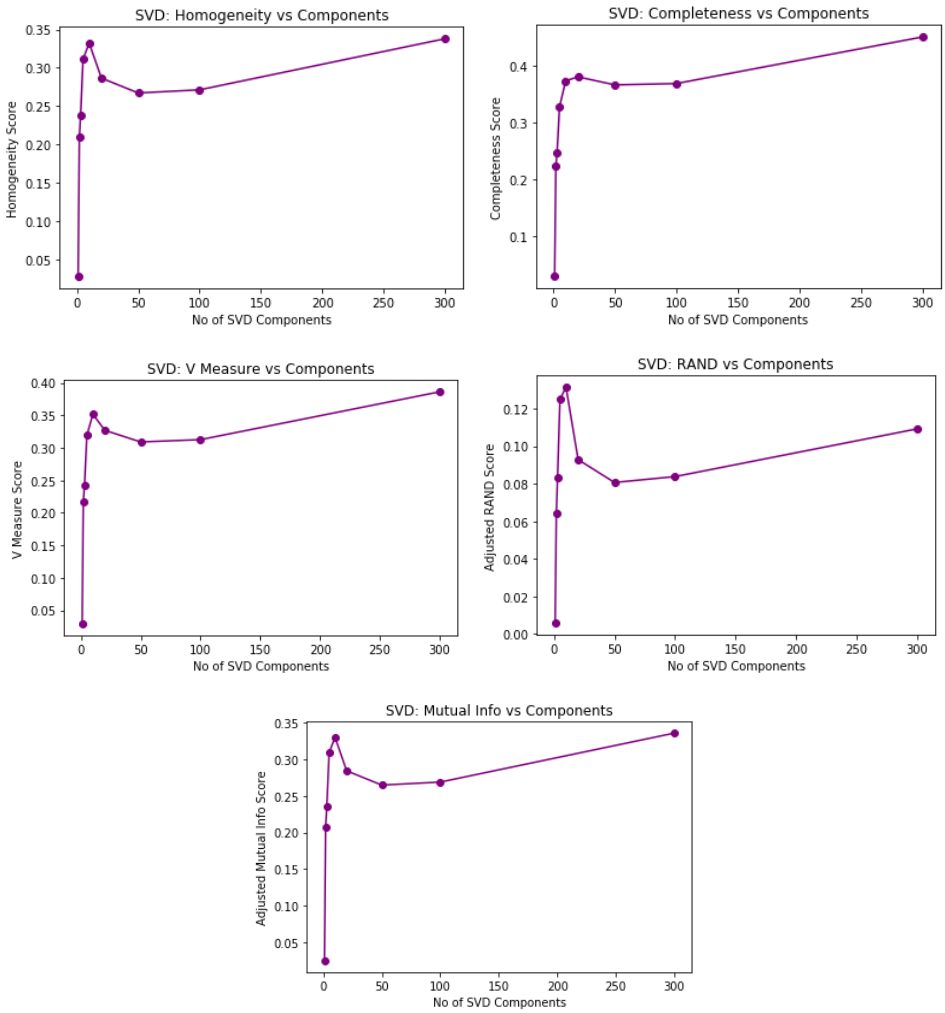

Figure 12.2: Measures of Scoring vs No of Components using SVD