ECE 219 Project 2: Clustering

Chuang Yu (305629107), Hongyi Chen (705186099), Zichun Chai (505625207)

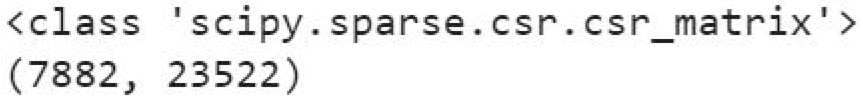
Question 1:

In this project, we explored the concept of clustering. Clustering algorithms are unsupervised methods for finding groups of data points that have similar representations in a feature space and it differs from classification that we did in the last project since it does not have any a priori labeling of data points.

In this project, we found proper representation of data, performed K-means, HDBscan, and DBscan on the dataset and evaluated the result of clustering. Lastly, we tried different preprocessing techniques that might increase the performance of clustering.

We worked with "20 Newsgroups" dataset that we already explored in Project 1. It is a collection of approximately 20,000 documents, partitioned (nearly) evenly across 20 different newsgroups, each corresponding to a different category (topic). Each topic can be viewed as a "class".

We followed the steps in Project 1, and transformed the documents into TF-IDF vectors. This measure takes into account count of words in the document, as normalized by a certain function of the frequency of the individual words in the whole corpus. Also, we used min_df = 3 and stop_words='english' in CountVectorizer without any stemming or lemmatization, and removed headers and footers in 20newsgroups. As for the 20newsgroups, we specified 8 categories and random_state to be 0. As a result, the dimensions of the TF-IDF matrix is:

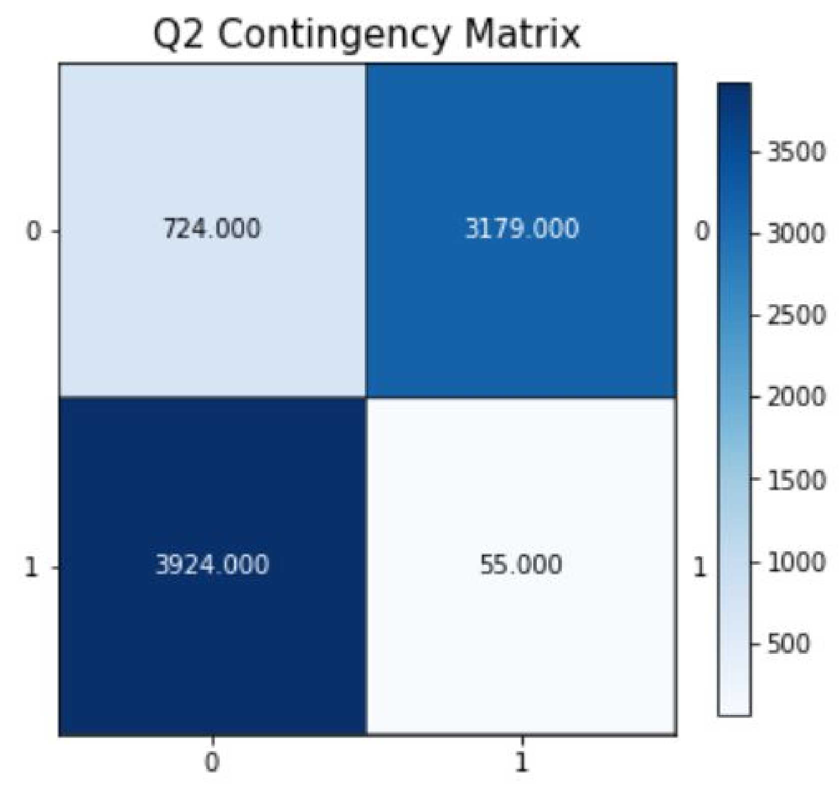


```
<class 'scipy.sparse.csr.csr_matrix'>
(7882, 23522)
```

Question 2:

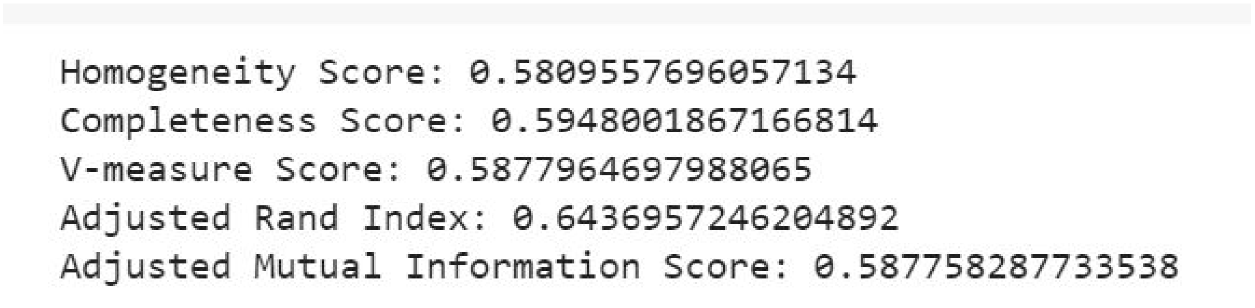
In this problem, we applied K-means clustering with k = 2 using the TF-IDF data from Q1. As for the parameters in the K-means class in sklearn, we specified that random_state=0, max_iter=1000 and n_init=30. The random_state parameter denoted the random number generation for centroid initialization. We needed to use an integer to make the randomness deterministic. The max_iter parameter represented the maximum number of iterations of the k-means algorithm for a single run. And the n_init parameter denoted the number of times that the k-means algorithm will be run with different centroid seeds. These parameters can improve the performance at a cost of computation power.

Given the clustering result and ground truth labels, contingency table A is the matrix whose entries Aij is the number of data points that belong to both the class Ci the cluster Kj. While plotting the contingency table, we utilized the provided plotmat.py file to visualize the matrix.



Question 3:

In order to evaluate clustering results, there are various measures for a given partition of the data points with respect to the ground truth. We adopted the measures homogeneity score, completeness score, V-measure, adjusted Rand score and adjusted mutual info score, all of which could be calculated by the corresponding functions provided in sklearn.metrics. Homogeneity is a measure of how "pure" the clusters are. On the other hand, a clustering result satisfies completeness if all data points of a class are assigned to the same cluster. And the V-measure is then defined to be the harmonic average of homogeneity score and completeness score. The adjusted Rand Index is similar to accuracy measure, which computes similarity between the clustering labels and ground truth labels. And the adjusted mutual information score measures the mutual information between the cluster label distribution and the ground truth label distributions. We have utilized these 5 measures for the K-means clustering results and the report is shown below:



```
Homogeneity Score: 0.5809557696057134
Completeness Score: 0.5948001867166814
V-measure Score: 0.5877964697988065
Adjusted Rand Index: 0.6436957246204892
Adjusted Mutual Information Score: 0.587758287733538
```

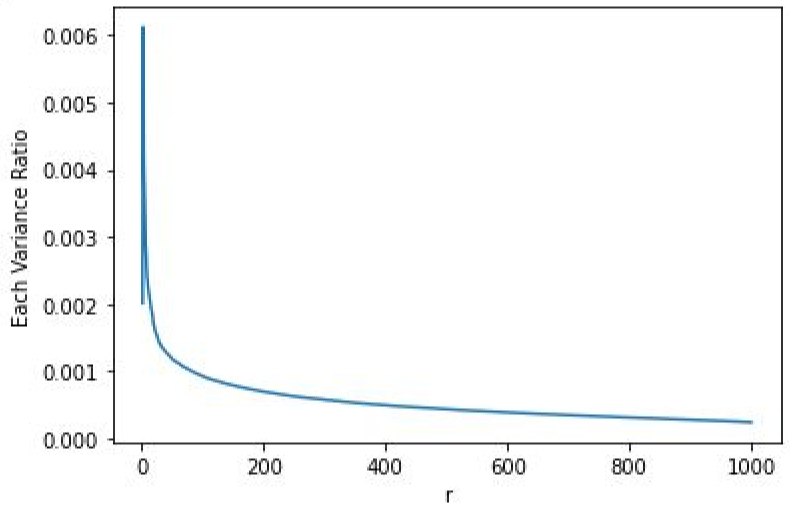
All the above 5 measures range between 0 and 1 and we wanted them to be as close to 1 as possible. Therefore, we would try different methods for dimensionality reduction, data transformation, and clustering later in this project.

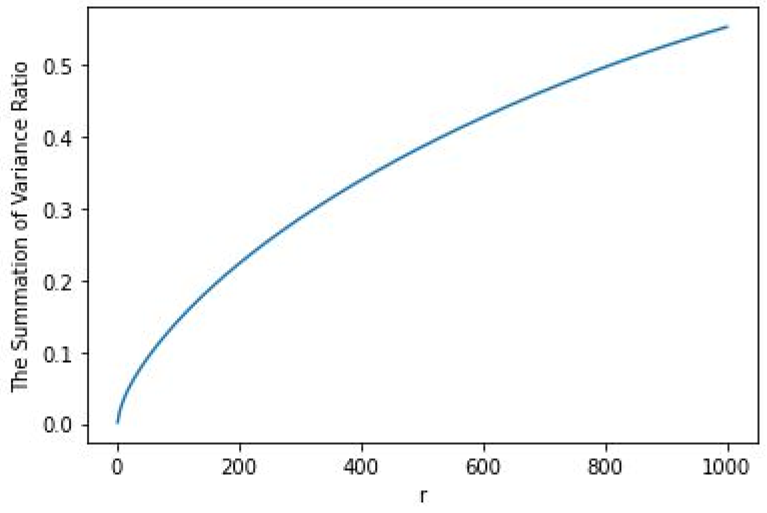
Question 4:

From the last problem we can see that high dimensional sparse TF-IDF vectors did not yield a good clustering result. One of the reasons is that in a high-dimensional space, the Euclidean distance is not a good metric anymore, in the sense that the distances between data points tends to be almost the same. Furthermore, K-means clustering is limited since it is good at round-shaped clusters without unequal variances. Therefore, in this problem, we tried to reduce the dimension of our data before clustering and applying other transformations.

Firstly, we investigated on what ratio of the variance of the original data is retained after the dimensionality reduction. Reconstructing the matrix with the truncated SVD representation, we plotted the percent of variance the top r principal components could retain vs r for r = 1 to 1000 by using explained_variance_ratio_ of TruncatedSVD objects.

The plots for both each variance ratio and the sum of variance ratio are shown below:



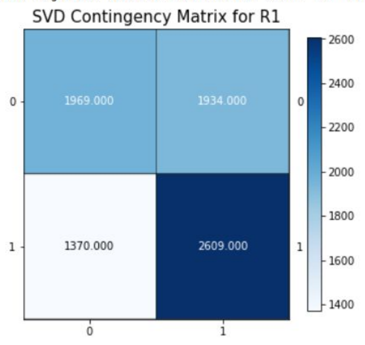


Question 5:

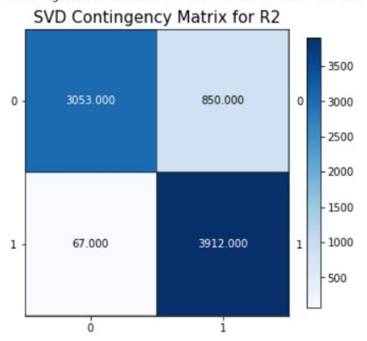
We used two methods to reduce the dimension of the data which were Singular Value Decomposition (SVD) and Non-negative Matrix Factorization (NMF). We swept over the dimension parameters r = [1, 2, 3, 5, 10, 20, 50, 100, 300] of each method, and based on the resulting homogeneity score, completeness score, V-measure, adjusted Rand score, adjusted mutual info score, and contingency matrix, we chose the the r that yielded the best results in terms of clustering purity metrics.

The results for SVD are shown below:

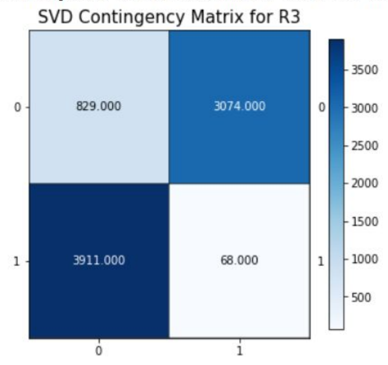
```
SVD Homogeneity Score for R1 0.019034591198472048
SVD Completeness Score for R1 0.019360460836506627
SVD V-measure Score for R1 0.01919614314371571
SVD Adjusted Rand Index for R1 0.02600277723478503
SVD Adjusted Mutual Information Score for R1 0.01910558756050995
```



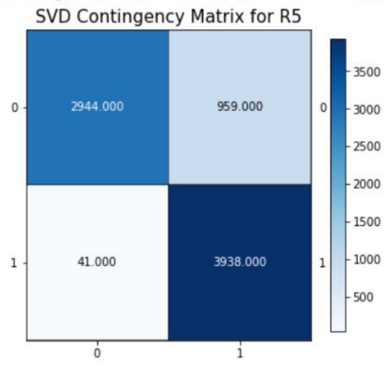
```
SVD Homogeneity Score for R2 0.5318527560716836
SVD Completeness Score for R2 0.5491344627077224
SVD V-measure Score for R2 0.5403554683557176
SVD Adjusted Rand Index for R2 0.588725244819119
SVD Adjusted Mutual Information Score for R2 0.5403127144748595
```



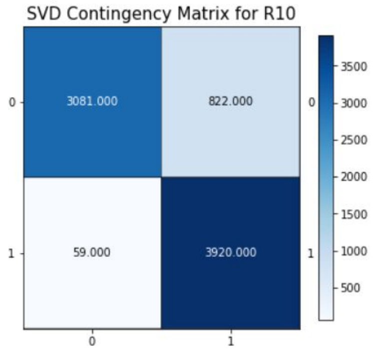
```
SVD Homogeneity Score for R3 0.5377651779879102
SVD Completeness Score for R3 0.5542779518154466
SVD V-measure Score for R3 0.5458967201533174
SVD Adjusted Rand Index for R3 0.5965400007596544
SVD Adjusted Mutual Information Score for R3 0.5458545177246293
```



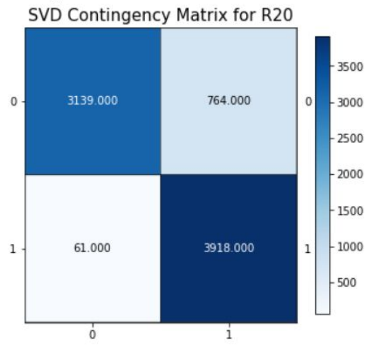
```
SVD Homogeneity Score for R5 0.517036422502093
SVD Completeness Score for R5 0.5401602853893641
SVD V-measure Score for R5 0.5283454620142469
SVD Adjusted Rand Index for R5 0.5568446089604012
SVD Adjusted Mutual Information Score for R5 0.5283013366507997
```



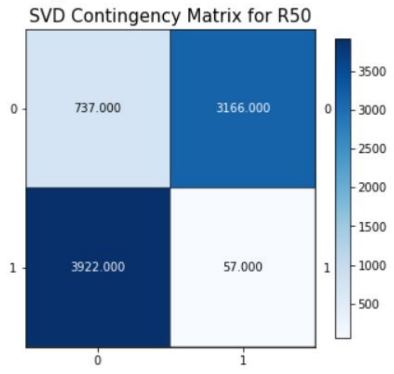
```
        0                1
SVD Homogeneity Score for R10 0.5460921220112308
SVD Completeness Score for R10 0.5629480411958336
SVD V-measure Score for R10 0.5543919879505773
SVD Adjusted Rand Index for R10 0.6028289245085969
SVD Adjusted Mutual Information Score for R10 0.5543505718649027
```



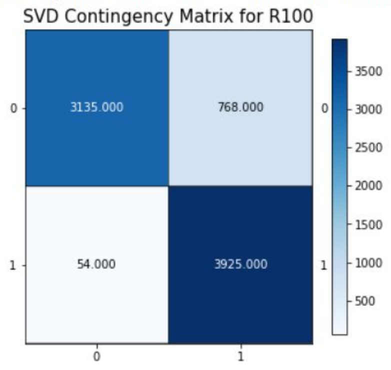
```
        0                1
SVD Homogeneity Score for R20 0.563426312706913
SVD Completeness Score for R20 0.5782222768022195
SVD V-measure Score for R20 0.5707284156217397
SVD Adjusted Rand Index for R20 0.6250997063911322
SVD Adjusted Mutual Information Score for R20 0.5706886059370165
```



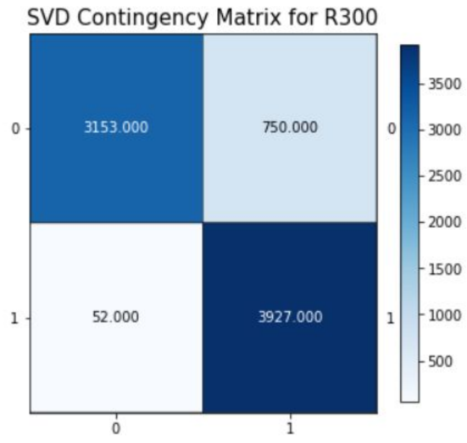
```
        0                1
SVD Homogeneity Score for R50 0.57516931569769
SVD Completeness Score for R50 0.5893200079451415
SVD V-measure Score for R50 0.5821586833212198
SVD Adjusted Rand Index for R50 0.6376018619358004
SVD Adjusted Mutual Information Score for R50 0.5821199646013246
```



```
SVD Homogeneity Score for R100 0.5670330934635874
SVD Completeness Score for R100 0.5823852599054565
SVD V-measure Score for R100 0.5746066513447411
SVD Adjusted Rand Index for R100 0.6263041993292062
SVD Adjusted Mutual Information Score for R100 0.5745671858723812
```

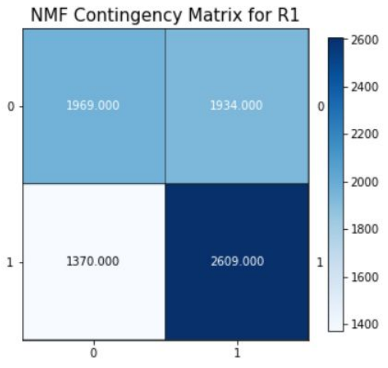


```
        0                1
SVD Homogeneity Score for R300 0.5744113629390132
SVD Completeness Score for R300 0.5892858802537997
SVD V-measure Score for R300 0.5817535576669169
SVD Adjusted Rand Index for R300 0.6343636646220453
SVD Adjusted Mutual Information Score for R300 0.5817147772526922
```

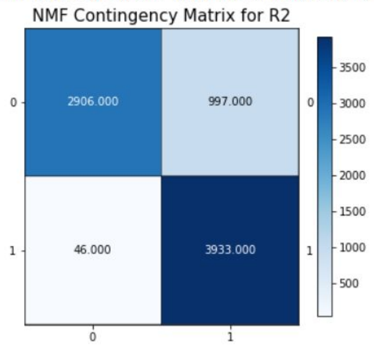


The results for NMF are shown below:

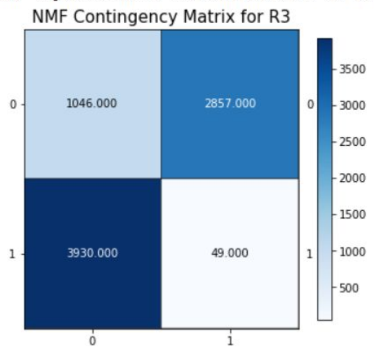
```
NMF Homogeneity Score for R1 0.019034591198472048
NMF Completeness Score for R1 0.019360460836506627
NMF V-measure Score for R1 0.01919614314371571
NMF Adjusted Rand Index for R1 0.02600277723478503
NMF Adjusted Mutual Information Score for R1 0.01910558756050995
```



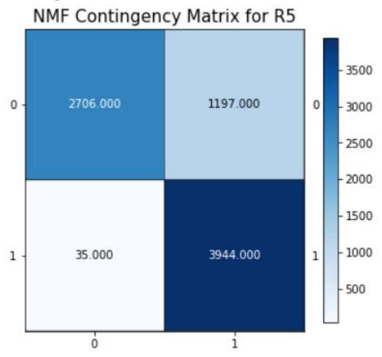
```
NMF Homogeneity Score for R2 0.5022478335120428
NMF Completeness Score for R2 0.5263843051392446
NMF V-measure Score for R2 0.5140328926482289
NMF Adjusted Rand Index for R2 0.5406769711846864
NMF Adjusted Mutual Information Score for R2 0.51398735739796
```



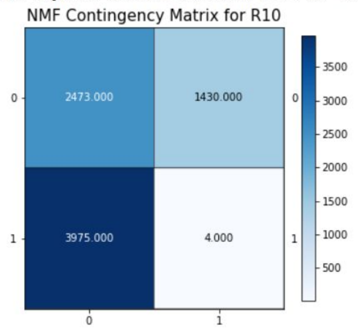
```
NMF Homogeneity Score for R3 0.48609683035120665
NMF Completeness Score for R3 0.5118300127229015
NMF V-measure Score for R3 0.49863163535474486
NMF Adjusted Rand Index for R3 0.521443534760384
NMF Adjusted Mutual Information Score for R3 0.4985845503268862
```



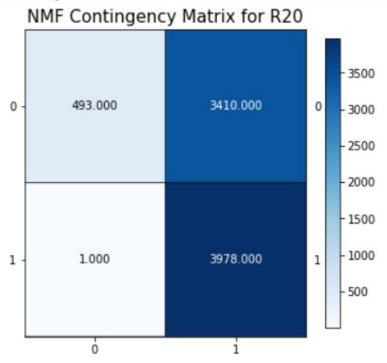
```
NMF Homogeneity Score for R5 0.45500660137619775
NMF Completeness Score for R5 0.4881474012058305
NMF V-measure Score for R5 0.47099474610769254
NMF Adjusted Rand Index for R5 0.4724383437991488
NMF Adjusted Mutual Information Score for R5 0.47094461230034934
```



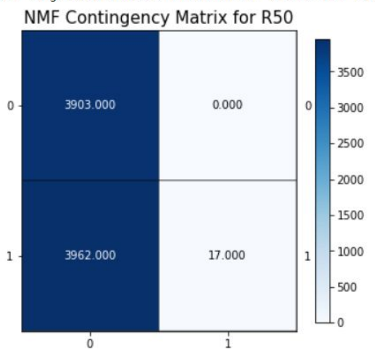
```
NMF Homogeneity Score for R10 0.20915909036595745
NMF Completeness Score for R10 0.30563865746695934
NMF V-measure Score for R10 0.248358132278426
NMF Adjusted Rand Index for R10 0.1378993331182748
NMF Adjusted Mutual Information Score for R10 0.24827640731705972
```



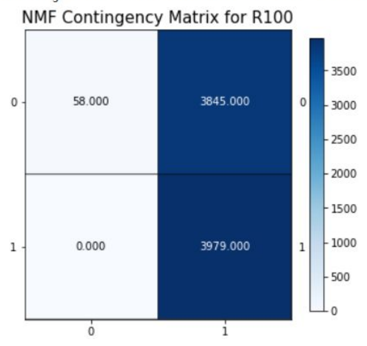
```
NMF Homogeneity Score for R20 0.06529404662847932
NMF Completeness Score for R20 0.1931811334155371
NMF V-measure Score for R20 0.09759991602155817
NMF Adjusted Rand Index for R20 0.0179867240535735
NMF Adjusted Mutual Information Score for R20 0.09747631727528892
```



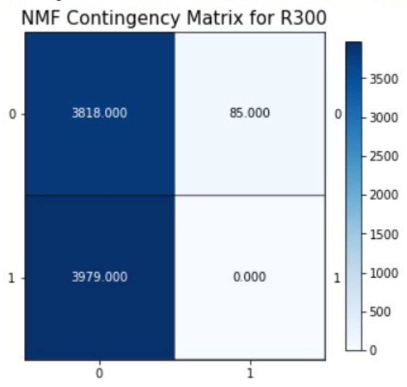
```
NMF Homogeneity Score for R50 0.0021303952018709594
NMF Completeness Score for R50 0.09591019434272455
NMF V-measure Score for R50 0.004168204593370178
NMF Adjusted Rand Index for R50 -6.487642756916324e-05
NMF Adjusted Mutual Information Score for R50 0.0039841010495616416
```



```
NMF Homogeneity Score for R100 0.007502018996706376
NMF Completeness Score for R100 0.11959893945551682
NMF V-measure Score for R100 0.014118438235357446
NMF Adjusted Rand Index for R100 0.0004994593822397486
NMF Adjusted Mutual Information Score for R100 0.01394709643405834
```



```
NMF Homogeneity Score for R300 0.011022010519163693
NMF Completeness Score for R300 0.12823289117096448
NMF V-measure Score for R300 0.02029923913966382
NMF Adjusted Rand Index for R300 0.000879750035061344
NMF Adjusted Mutual Information Score for R300 0.02013308476943905
```



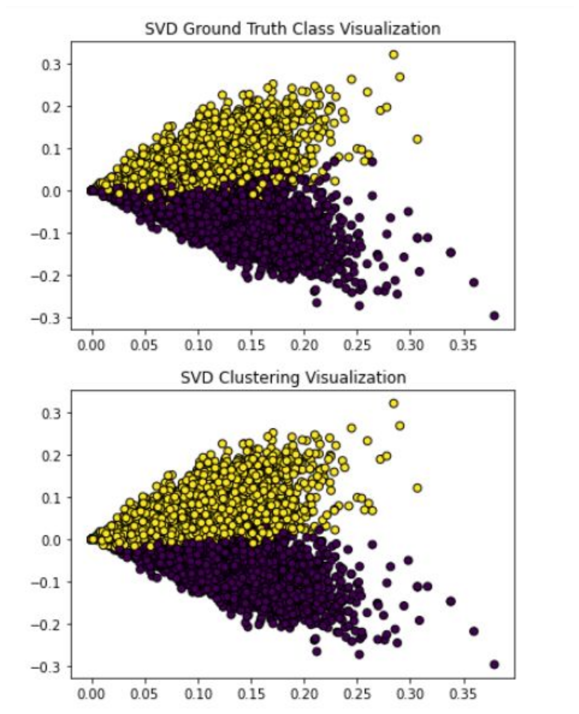
From the results shown above, we could tell that r = 50 is good for SVD, and r = 2 is good for NMF, and we also observed that there is a trade-off between the information preservation, and better performance of K-means in lower dimensions. So we would use these two values in the following questions as the best case.

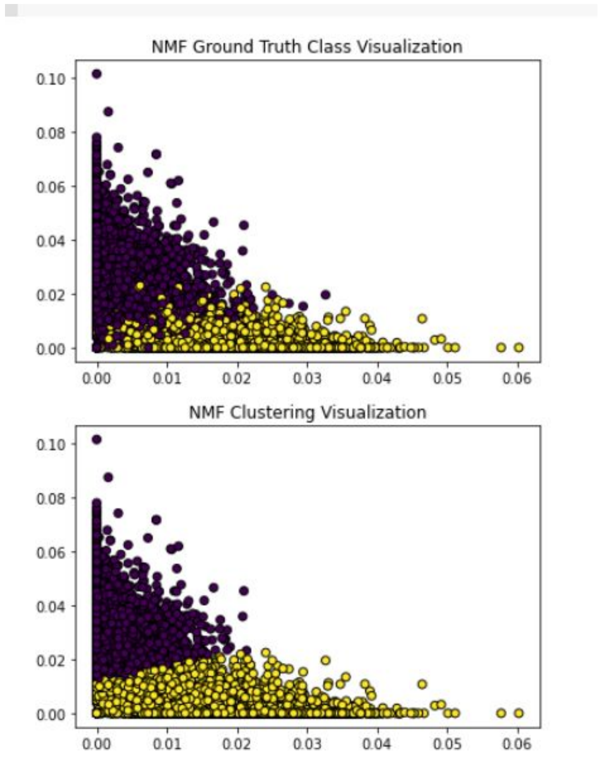
Question 6:

We did observe this non-monotonic behavior of the measures as we increased r. There are mainly two reasons for this. Firstly, if r is too small, we do not have enough information for the TF-IDF vectors and the later clustering process. Secondly, if r is too big, high dimensional sparse TF-IDF vectors do not yield a good clustering result because the Euclidean distance tends to be almost the same in a high-dimensional space. These two effects would greatly harm the performance of K-means clustering and lead to the non-monotonic behavior.

Question 7:

As for visualization, we could visualize the results by projecting the dim-reduced data points onto a 2-D plane with SVD, and coloring the points according to ground truth class label and clustering label respectively. We chose r = 50 for SVD and r = 2 for NMF from the results that we have derived from Q5. The plots are shown below:



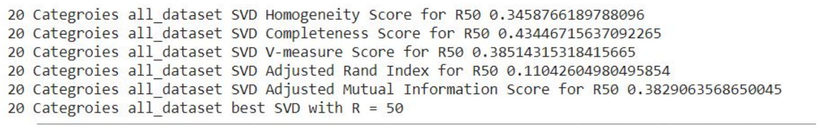


Question 8:

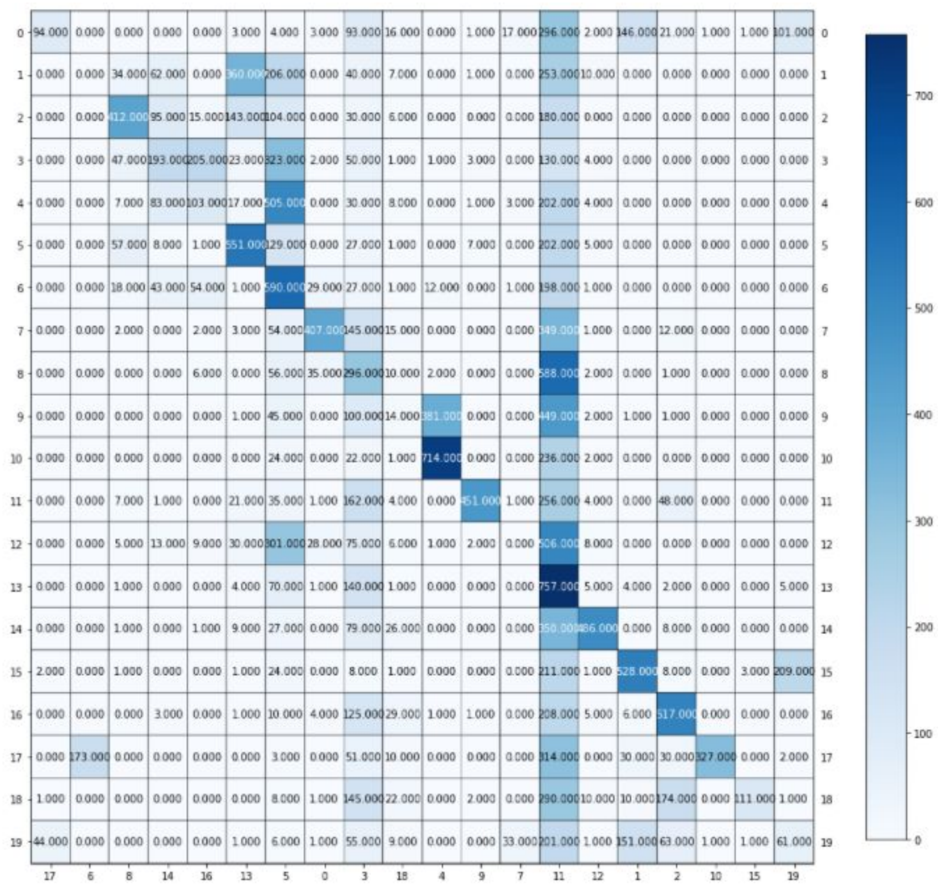
Generally speaking, we observed a boundary between two classes and they were separated clearly. And the ground truth class label plots share a lot in common with the clustering label plots in both dimensionality reduction techniques. However, we still observed some outliers, and therefore we would still need to improve our techniques in dimensionality reduction and clustering for better performance.

Question 9:

In the previous questions, we have been dealing with a relatively simple clustering task with only two well-separated classes “comp” and “rec” for only 8 categories. In this problem, we would cluster for the entire 20 categories in the 20newsgroups dataset. We loaded the documents with the same configuration as in Q1 but for all the 20 categories. We utilized CountVectorizer and TF-IDF for feature extraction, and SVD (r = 50) for dimensionality reduction, and at last performed K-means clustering with n_components = 20. Also, there is a mismatch between cluster labels and class labels, and as a result, the high-value entries of the 20 * 20 contingency matrix can be scattered around, making it messy to inspect, even if the clustering result is not bad. Therefore, we used scipy.optimize.linear_sum_assignment to identify the best-matching cluster-class pairs, and permute the columns of the contingency matrix accordingly. The contingency matrix and the five clustering metrics are shown below:



```
20 Categroies all_dataset SVD Homogeneity Score for R50 0.3458766189788096
20 Categroies all_dataset SVD Completeness Score for R50 0.43446715637092265
20 Categroies all_dataset SVD V-measure Score for R50 0.38514315318415665
20 Categroies all_dataset SVD Adjusted Rand Index for R50 0.11042604980495854
20 Categroies all_dataset SVD Adjusted Mutual Information Score for R50 0.3829063568650045
20 Categroies all_dataset best SVD with R = 50
```

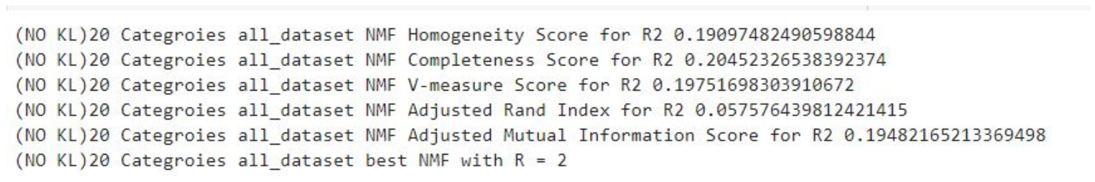


	17	6	8	14	16	13	5	0	3	18	4	9	7	11	12	1	2	10	15	19
0	94.000	0.000	0.000	0.000	0.000	3.000	4.000	3.000	93.000	16.000	0.000	1.000	17.000	296.000	2.000	146.000	21.000	1.000	1.000	101.000
1	0.000	0.000	34.000	62.000	0.000	360.000	206.000	0.000	40.000	7.000	0.000	1.000	0.000	253.000	10.000	0.000	0.000	0.000	0.000	0.000
2	0.000	0.000	412.000	95.000	15.000	143.000	104.000	0.000	30.000	6.000	0.000	0.000	0.000	180.000	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	47.000	193.000	205.000	23.000	323.000	2.000	50.000	1.000	1.000	3.000	0.000	130.000	4.000	0.000	0.000	0.000	0.000	0.000
4	0.000	0.000	7.000	83.000	103.000	17.000	505.000	0.000	30.000	8.000	0.000	1.000	3.000	202.000	4.000	0.000	0.000	0.000	0.000	0.000
5	0.000	0.000	57.000	8.000	1.000	551.000	129.000	0.000	27.000	1.000	0.000	7.000	0.000	202.000	5.000	0.000	0.000	0.000	0.000	0.000
6	0.000	0.000	18.000	43.000	54.000	1.000	590.000	29.000	27.000	1.000	12.000	0.000	1.000	198.000	1.000	0.000	0.000	0.000	0.000	0.000
7	0.000	0.000	2.000	0.000	2.000	3.000	54.000	407.000	145.000	15.000	0.000	0.000	0.000	349.000	1.000	0.000	12.000	0.000	0.000	0.000
8	0.000	0.000	0.000	0.000	6.000	0.000	56.000	35.000	296.000	10.000	2.000	0.000	0.000	588.000	2.000	0.000	1.000	0.000	0.000	0.000
9	0.000	0.000	0.000	0.000	0.000	1.000	45.000	0.000	100.000	14.000	381.000	0.000	0.000	449.000	2.000	1.000	1.000	0.000	0.000	0.000
10	0.000	0.000	0.000	0.000	0.000	0.000	24.000	0.000	22.000	1.000	714.000	0.000	0.000	236.000	2.000	0.000	0.000	0.000	0.000	0.000
11	0.000	0.000	7.000	1.000	0.000	21.000	35.000	1.000	162.000	4.000	0.000	451.000	1.000	256.000	4.000	0.000	48.000	0.000	0.000	0.000
12	0.000	0.000	5.000	13.000	9.000	30.000	301.000	28.000	75.000	6.000	1.000	2.000	0.000	506.000	8.000	0.000	0.000	0.000	0.000	0.000
13	0.000	0.000	1.000	0.000	0.000	4.000	70.000	1.000	140.000	1.000	0.000	0.000	0.000	757.000	5.000	4.000	2.000	0.000	0.000	5.000
14	0.000	0.000	1.000	0.000	1.000	9.000	27.000	0.000	79.000	26.000	0.000	0.000	0.000	350.000	486.000	0.000	8.000	0.000	0.000	0.000
15	2.000	0.000	1.000	0.000	0.000	1.000	24.000	0.000	8.000	1.000	0.000	0.000	0.000	211.000	1.000	528.000	8.000	0.000	3.000	209.000
16	0.000	0.000	0.000	3.000	0.000	1.000	10.000	4.000	125.000	29.000	1.000	1.000	0.000	208.000	5.000	6.000	517.000	0.000	0.000	0.000
17	0.000	173.000	0.000	0.000	0.000	0.000	3.000	0.000	51.000	10.000	0.000	0.000	0.000	314.000	0.000	30.000	30.000	327.000	0.000	2.000
18	1.000	0.000	0.000	0.000	0.000	0.000	8.000	1.000	145.000	22.000	0.000	2.000	0.000	290.000	10.000	10.000	174.000	0.000	111.000	1.000
19	44.000	0.000	0.000	0.000	0.000	1.000	6.000	1.000	55.000	9.000	0.000	0.000	33.000	201.000	1.000	151.000	63.000	1.000	1.000	61.000

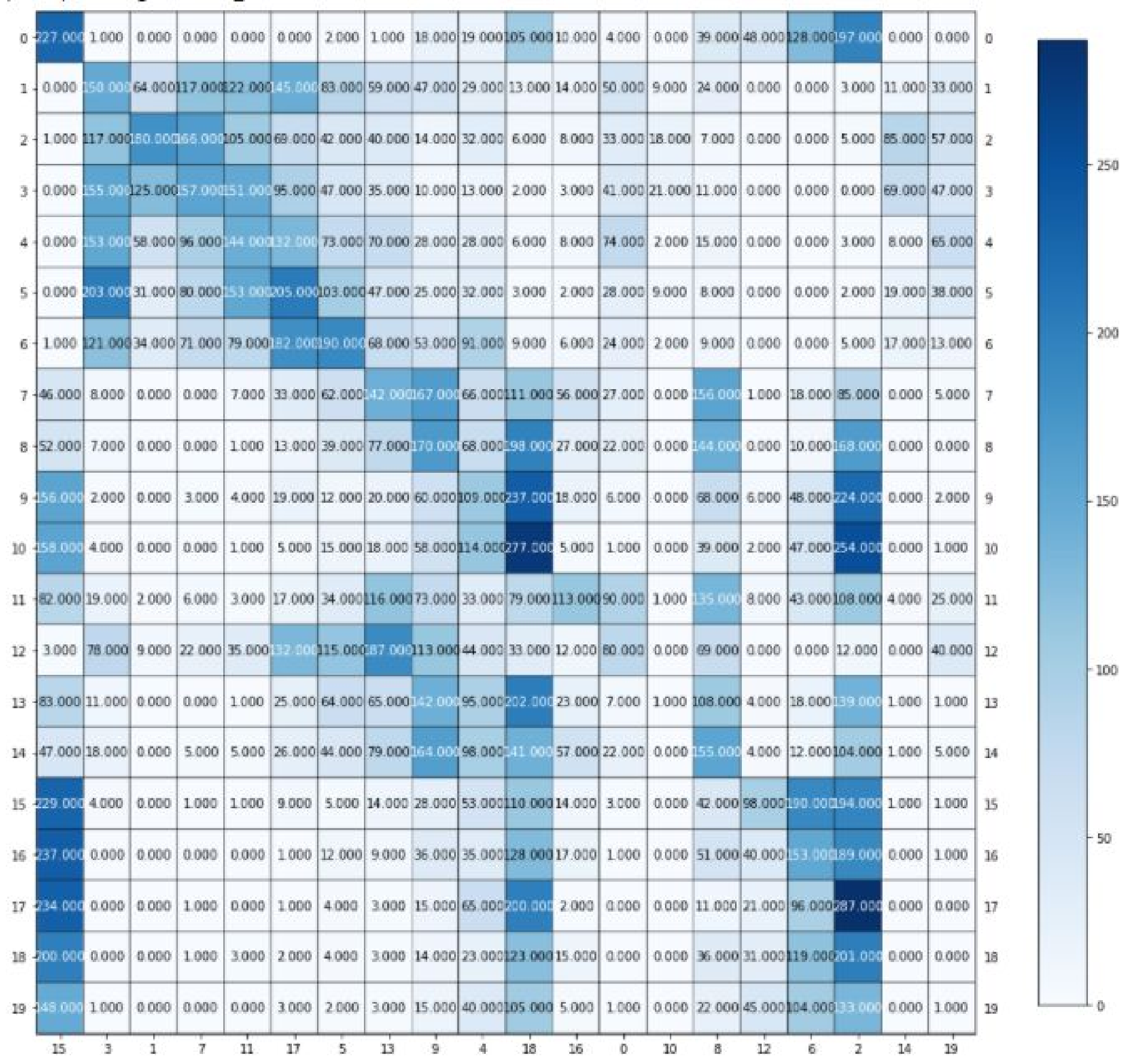
Question 10:

NMF uses Frobenius norm as its cost function by default. And we decided to try another choice which was Kullback-Leibler divergence by specifying beta_loss = "kullback-leibler", solver = "mu", and observed whether it helped with the clustering of our text data. The results for both case with and without KL are shown below:

Without KL:



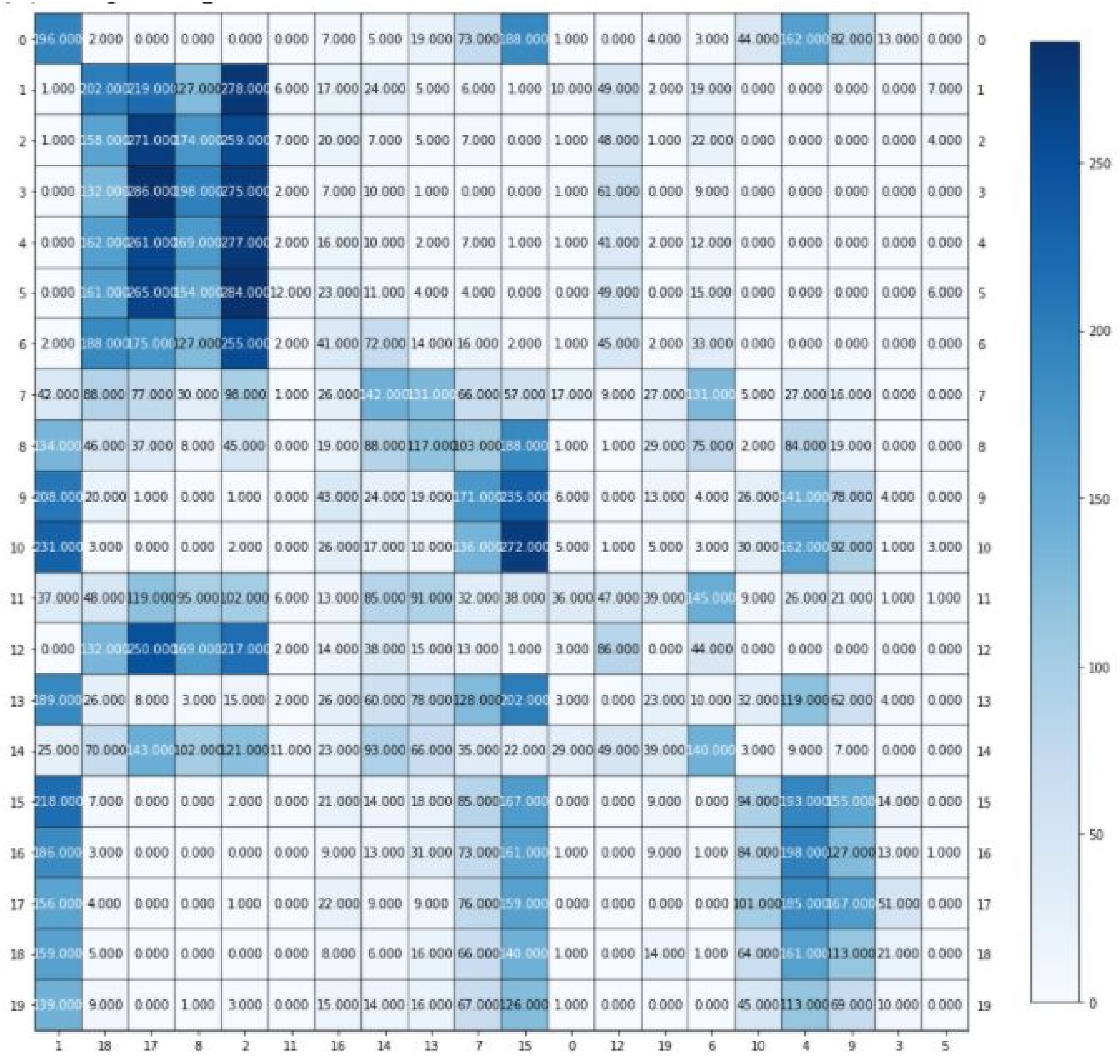
```
(NO KL)20 Categroies all_dataset NMF Homogeneity Score for R2 0.19097482490598844
(NO KL)20 Categroies all_dataset NMF Completeness Score for R2 0.20452326538392374
(NO KL)20 Categroies all_dataset NMF V-measure Score for R2 0.19751698303910672
(NO KL)20 Categroies all_dataset NMF Adjusted Rand Index for R2 0.057576439812421415
(NO KL)20 Categroies all_dataset NMF Adjusted Mutual Information Score for R2 0.19482165213369498
(NO KL)20 Categroies all_dataset best NMF with R = 2
```



	15	3	1	7	11	17	5	13	9	4	18	16	0	10	8	12	6	2	14	19
0	227.000	1.000	0.000	0.000	0.000	0.000	2.000	1.000	18.000	19.000	105.000	10.000	4.000	0.000	39.000	48.000	128.000	197.000	0.000	0.000
1	0.000	150.000	64.000	117.000	122.000	145.000	83.000	59.000	47.000	29.000	13.000	14.000	50.000	9.000	24.000	0.000	0.000	3.000	11.000	33.000
2	1.000	117.000	180.000	166.000	105.000	69.000	42.000	40.000	14.000	32.000	6.000	8.000	33.000	18.000	7.000	0.000	0.000	5.000	85.000	57.000
3	0.000	155.000	125.000	157.000	151.000	95.000	47.000	35.000	10.000	13.000	2.000	3.000	41.000	21.000	11.000	0.000	0.000	0.000	69.000	47.000
4	0.000	153.000	58.000	96.000	144.000	132.000	73.000	70.000	28.000	28.000	6.000	8.000	74.000	2.000	15.000	0.000	0.000	3.000	8.000	65.000
5	0.000	203.000	31.000	80.000	153.000	205.000	103.000	47.000	25.000	32.000	3.000	2.000	28.000	9.000	8.000	0.000	0.000	2.000	19.000	38.000
6	1.000	121.000	34.000	71.000	79.000	182.000	190.000	68.000	53.000	91.000	9.000	6.000	24.000	2.000	9.000	0.000	0.000	5.000	17.000	13.000
7	46.000	8.000	0.000	0.000	7.000	33.000	62.000	142.000	167.000	66.000	111.000	56.000	27.000	0.000	156.000	1.000	18.000	85.000	0.000	5.000
8	52.000	7.000	0.000	0.000	1.000	13.000	39.000	77.000	170.000	68.000	198.000	27.000	22.000	0.000	144.000	0.000	10.000	168.000	0.000	0.000
9	156.000	2.000	0.000	3.000	4.000	19.000	12.000	20.000	60.000	109.000	237.000	18.000	6.000	0.000	68.000	6.000	48.000	224.000	0.000	2.000
10	158.000	4.000	0.000	0.000	1.000	5.000	15.000	18.000	58.000	114.000	277.000	5.000	1.000	0.000	39.000	2.000	47.000	254.000	0.000	1.000
11	82.000	19.000	2.000	6.000	3.000	17.000	34.000	116.000	73.000	33.000	79.000	113.000	90.000	1.000	135.000	8.000	43.000	108.000	4.000	25.000
12	3.000	78.000	9.000	22.000	35.000	132.000	115.000	187.000	113.000	44.000	33.000	12.000	80.000	0.000	69.000	0.000	0.000	12.000	0.000	40.000
13	83.000	11.000	0.000	0.000	1.000	25.000	64.000	65.000	142.000	95.000	202.000	23.000	7.000	1.000	108.000	4.000	18.000	139.000	1.000	1.000
14	47.000	18.000	0.000	5.000	5.000	26.000	44.000	79.000	164.000	98.000	141.000	57.000	22.000	0.000	155.000	4.000	12.000	104.000	1.000	5.000
15	229.000	4.000	0.000	1.000	1.000	9.000	5.000	14.000	28.000	53.000	110.000	14.000	3.000	0.000	42.000	98.000	190.000	194.000	1.000	1.000
16	237.000	0.000	0.000	0.000	0.000	1.000	12.000	9.000	36.000	35.000	128.000	17.000	1.000	0.000	51.000	40.000	153.000	189.000	0.000	1.000
17	234.000	0.000	0.000	1.000	0.000	1.000	4.000	3.000	15.000	65.000	200.000	2.000	0.000	0.000	11.000	21.000	96.000	287.000	0.000	0.000
18	200.000	0.000	0.000	1.000	3.000	2.000	4.000	3.000	14.000	23.000	123.000	15.000	0.000	0.000	36.000	31.000	119.000	201.000	0.000	0.000
19	148.000	1.000	0.000	0.000	0.000	3.000	2.000	3.000	15.000	40.000	105.000	5.000	1.000	0.000	22.000	45.000	104.000	133.000	0.000	1.000

With KL:

```
(KL)20 Categroies all_dataset NMF Homogeneity Score for R2 0.20217506002873892
(KL)20 Categroies all_dataset NMF Completeness Score for R2 0.2260432455956135
(KL)20 Categroies all_dataset NMF V-measure Score for R2 0.21344396606656976
(KL)20 Categroies all_dataset NMF Adjusted Rand Index for R2 0.06566950852631316
(KL)20 Categroies all_dataset NMF Adjusted Mutual Information Score for R2 0.21071407416901536
(KL)20 Categroies all_dataset best NMF with R = 2
```



	1	18	17	8	2	11	16	14	13	7	15	0	12	19	6	10	4	9	3	5	
0	196.000	2.000	0.000	0.000	0.000	0.000	7.000	5.000	19.000	73.000	188.000	1.000	0.000	4.000	3.000	44.000	162.000	82.000	13.000	0.000	0
1	1.000	202.000	219.000	127.000	278.000	6.000	17.000	24.000	5.000	6.000	1.000	10.000	49.000	2.000	19.000	0.000	0.000	0.000	0.000	7.000	1
2	1.000	158.000	271.000	174.000	259.000	7.000	20.000	7.000	5.000	7.000	0.000	1.000	48.000	1.000	22.000	0.000	0.000	0.000	0.000	4.000	2
3	0.000	132.000	286.000	198.000	275.000	2.000	7.000	10.000	1.000	0.000	0.000	1.000	61.000	0.000	9.000	0.000	0.000	0.000	0.000	0.000	3
4	0.000	162.000	261.000	169.000	277.000	2.000	16.000	10.000	2.000	7.000	1.000	1.000	41.000	2.000	12.000	0.000	0.000	0.000	0.000	0.000	4
5	0.000	161.000	265.000	154.000	284.000	12.000	23.000	11.000	4.000	4.000	0.000	0.000	49.000	0.000	15.000	0.000	0.000	0.000	0.000	6.000	5
6	2.000	188.000	175.000	127.000	255.000	2.000	41.000	72.000	14.000	16.000	2.000	1.000	45.000	2.000	33.000	0.000	0.000	0.000	0.000	0.000	6
7	42.000	88.000	77.000	30.000	98.000	1.000	26.000	142.000	131.000	66.000	57.000	17.000	9.000	27.000	131.000	5.000	27.000	16.000	0.000	0.000	7
8	134.000	46.000	37.000	8.000	45.000	0.000	19.000	88.000	117.000	103.000	188.000	1.000	1.000	29.000	75.000	2.000	84.000	19.000	0.000	0.000	8
9	208.000	20.000	1.000	0.000	1.000	0.000	43.000	24.000	19.000	171.000	235.000	6.000	0.000	13.000	4.000	26.000	141.000	78.000	4.000	0.000	9
10	231.000	3.000	0.000	0.000	2.000	0.000	26.000	17.000	10.000	136.000	272.000	5.000	1.000	5.000	3.000	30.000	162.000	92.000	1.000	3.000	10
11	37.000	48.000	119.000	95.000	102.000	6.000	13.000	85.000	91.000	32.000	38.000	36.000	47.000	39.000	145.000	9.000	26.000	21.000	1.000	1.000	11
12	0.000	132.000	250.000	169.000	217.000	2.000	14.000	38.000	15.000	13.000	1.000	3.000	86.000	0.000	44.000	0.000	0.000	0.000	0.000	0.000	12
13	189.000	26.000	8.000	3.000	15.000	2.000	26.000	60.000	78.000	128.000	202.000	3.000	0.000	23.000	10.000	32.000	119.000	62.000	4.000	0.000	13
14	25.000	70.000	143.000	102.000	121.000	11.000	23.000	93.000	66.000	35.000	22.000	29.000	49.000	39.000	140.000	3.000	9.000	7.000	0.000	0.000	14
15	218.000	7.000	0.000	0.000	2.000	0.000	21.000	14.000	18.000	85.000	167.000	0.000	0.000	9.000	0.000	94.000	193.000	155.000	14.000	0.000	15
16	186.000	3.000	0.000	0.000	0.000	0.000	9.000	13.000	31.000	73.000	161.000	1.000	0.000	9.000	1.000	84.000	198.000	127.000	13.000	1.000	16
17	156.000	4.000	0.000	0.000	1.000	0.000	22.000	9.000	9.000	76.000	159.000	0.000	0.000	0.000	0.000	101.000	185.000	167.000	51.000	0.000	17
18	159.000	5.000	0.000	0.000	0.000	0.000	8.000	6.000	16.000	66.000	140.000	1.000	0.000	14.000	1.000	64.000	161.000	113.000	21.000	0.000	18
19	139.000	9.000	0.000	1.000	3.000	0.000	15.000	14.000	16.000	67.000	126.000	1.000	0.000	0.000	0.000	45.000	113.000	69.000	10.000	0.000	19

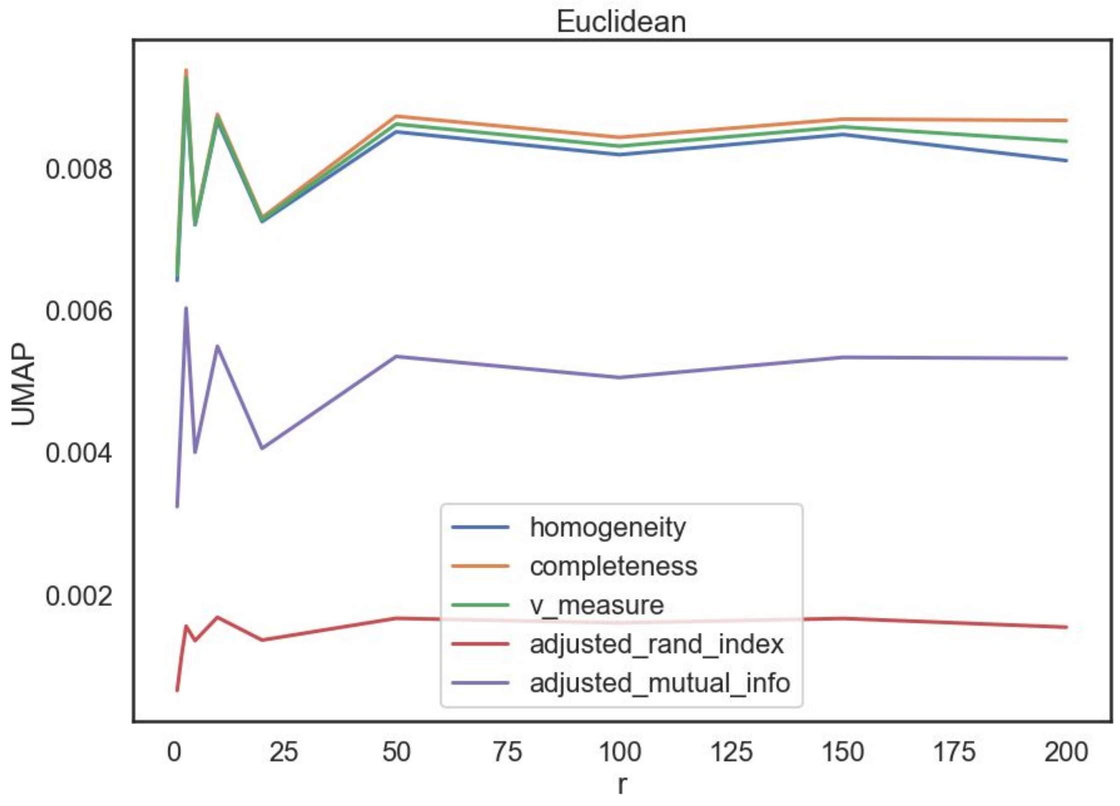
From the results we could see that the case with Kullback-Leibler divergence has a better performance than the case without KL divergence in all five metrics. Therefore, it is clear that

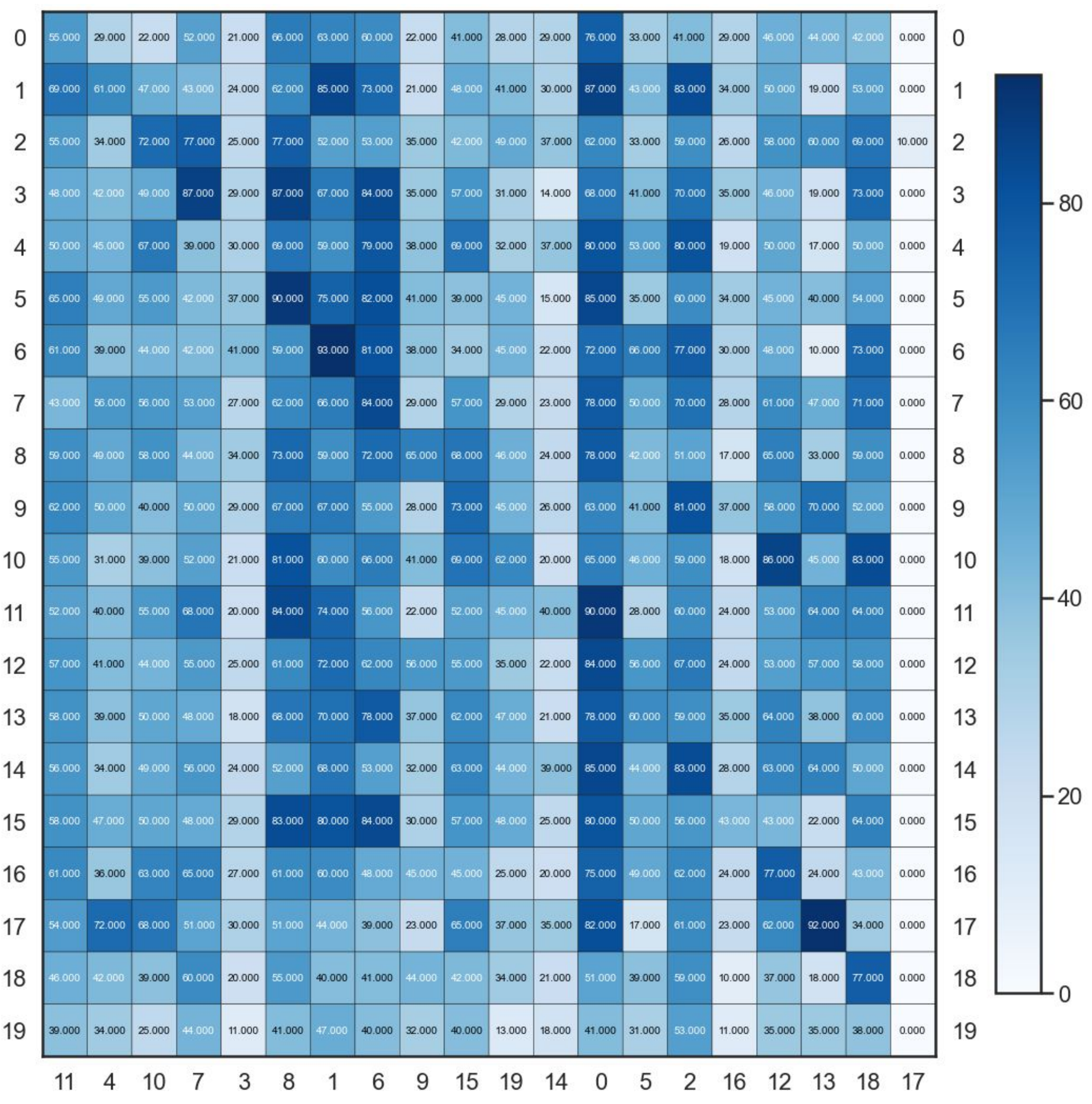
Kullback-Leibler divergence is a better cost function than Frobenius norm because it improves the performance.

Question 11:

In this question we used UMAP to reduce the dimensionality of the 20 categories of the TF-IDF matrix with two settings of metric = ‘euclidean’ and metric = ‘cosine’. The plots below show the results for metric = ‘euclidean’. Best n_component is 3.

homogeneity_score 0.008888883402648982
completeness_score 0.009163902973211941
v_measure_score 0.00902429833778883
adjusted_rand_score 0.0014956927303962568
adjusted_mutual_info_score 0.005773132022734455
Contingency matrix with Euclidean





	11	4	10	7	3	8	1	6	9	15	19	14	0	5	2	16	12	13	18	17	
0	55.000	29.000	22.000	52.000	21.000	66.000	63.000	60.000	22.000	41.000	28.000	29.000	76.000	33.000	41.000	29.000	46.000	44.000	42.000	0.000	0
1	69.000	61.000	47.000	43.000	24.000	62.000	85.000	73.000	21.000	48.000	41.000	30.000	87.000	43.000	83.000	34.000	50.000	19.000	53.000	0.000	1
2	55.000	34.000	72.000	77.000	25.000	77.000	52.000	53.000	35.000	42.000	49.000	37.000	62.000	33.000	59.000	26.000	58.000	60.000	69.000	10.000	2
3	48.000	42.000	49.000	87.000	29.000	87.000	67.000	84.000	35.000	57.000	31.000	14.000	68.000	41.000	70.000	35.000	46.000	19.000	73.000	0.000	3
4	50.000	45.000	67.000	39.000	30.000	69.000	59.000	79.000	38.000	69.000	32.000	37.000	80.000	53.000	80.000	19.000	50.000	17.000	50.000	0.000	4
5	65.000	49.000	55.000	42.000	37.000	90.000	75.000	82.000	41.000	39.000	45.000	15.000	85.000	35.000	60.000	34.000	45.000	40.000	54.000	0.000	5
6	61.000	39.000	44.000	42.000	41.000	59.000	93.000	81.000	38.000	34.000	45.000	22.000	72.000	66.000	77.000	30.000	48.000	10.000	73.000	0.000	6
7	43.000	56.000	56.000	53.000	27.000	62.000	66.000	84.000	29.000	57.000	29.000	23.000	78.000	50.000	70.000	28.000	61.000	47.000	71.000	0.000	7
8	59.000	49.000	58.000	44.000	34.000	73.000	59.000	72.000	65.000	68.000	46.000	24.000	78.000	42.000	51.000	17.000	65.000	33.000	59.000	0.000	8
9	62.000	50.000	40.000	50.000	29.000	67.000	67.000	55.000	28.000	73.000	45.000	26.000	63.000	41.000	81.000	37.000	58.000	70.000	52.000	0.000	9
10	55.000	31.000	39.000	52.000	21.000	81.000	60.000	66.000	41.000	69.000	62.000	20.000	65.000	46.000	59.000	18.000	86.000	45.000	83.000	0.000	10
11	52.000	40.000	55.000	68.000	20.000	84.000	74.000	56.000	22.000	52.000	45.000	40.000	90.000	28.000	60.000	24.000	53.000	64.000	64.000	0.000	11
12	57.000	41.000	44.000	55.000	25.000	61.000	72.000	62.000	56.000	55.000	35.000	22.000	84.000	56.000	67.000	24.000	53.000	57.000	58.000	0.000	12
13	58.000	39.000	50.000	48.000	18.000	68.000	70.000	78.000	37.000	62.000	47.000	21.000	78.000	60.000	59.000	35.000	64.000	38.000	60.000	0.000	13
14	56.000	34.000	49.000	56.000	24.000	52.000	68.000	53.000	32.000	63.000	44.000	39.000	85.000	44.000	83.000	28.000	63.000	64.000	50.000	0.000	14
15	58.000	47.000	50.000	48.000	29.000	83.000	80.000	84.000	30.000	57.000	48.000	25.000	80.000	50.000	56.000	43.000	43.000	22.000	64.000	0.000	15
16	61.000	36.000	63.000	65.000	27.000	61.000	60.000	48.000	45.000	45.000	25.000	20.000	75.000	49.000	62.000	24.000	77.000	24.000	43.000	0.000	16
17	54.000	72.000	68.000	51.000	30.000	51.000	44.000	39.000	23.000	65.000	37.000	35.000	82.000	17.000	61.000	23.000	62.000	92.000	34.000	0.000	17
18	46.000	42.000	39.000	60.000	20.000	55.000	40.000	41.000	44.000	42.000	34.000	21.000	51.000	39.000	59.000	10.000	37.000	18.000	77.000	0.000	18
19	39.000	34.000	25.000	44.000	11.000	41.000	47.000	40.000	32.000	40.000	13.000	18.000	41.000	31.000	53.000	11.000	35.000	35.000	38.000	0.000	19

cos

Best n_component is 50,

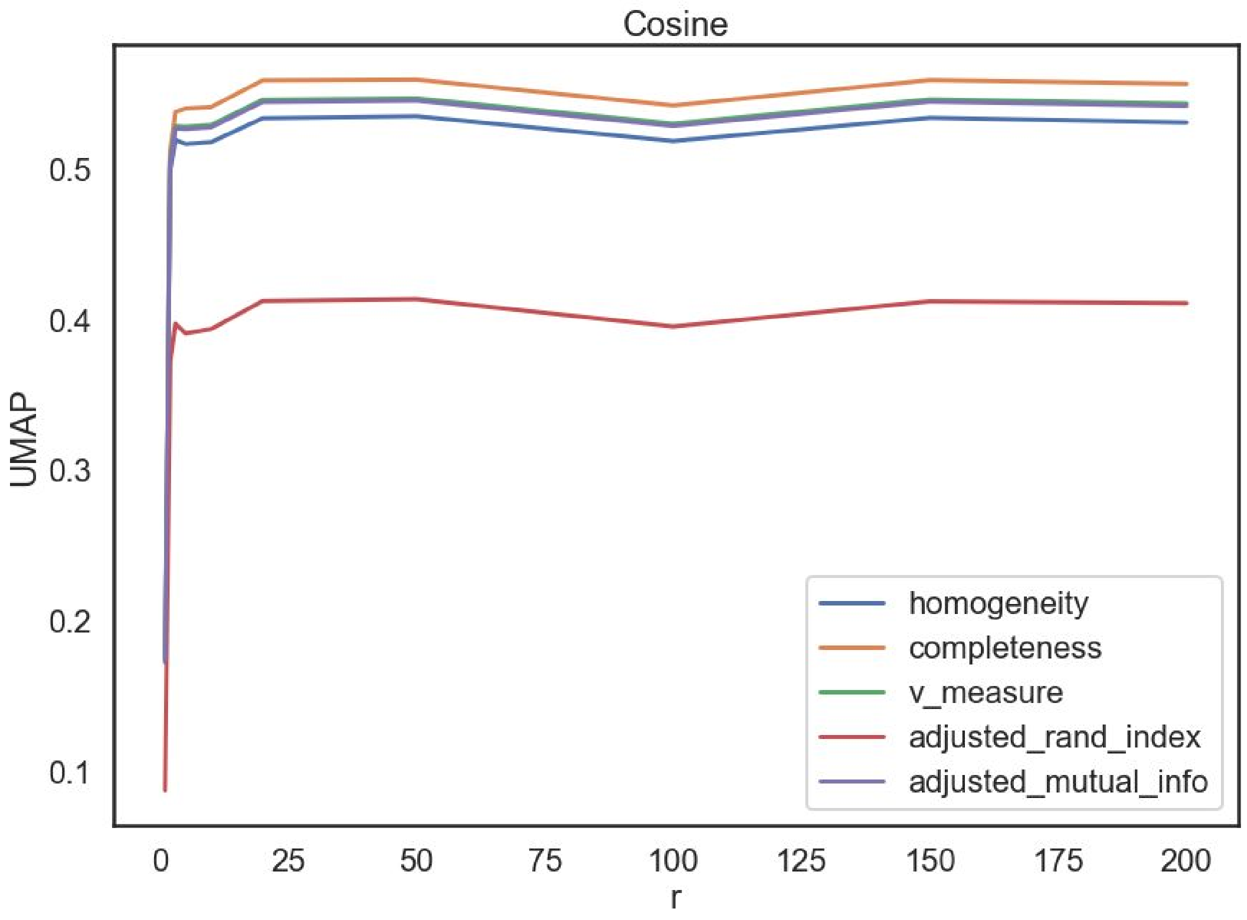
Homogeneity Score 0.5360492927573465

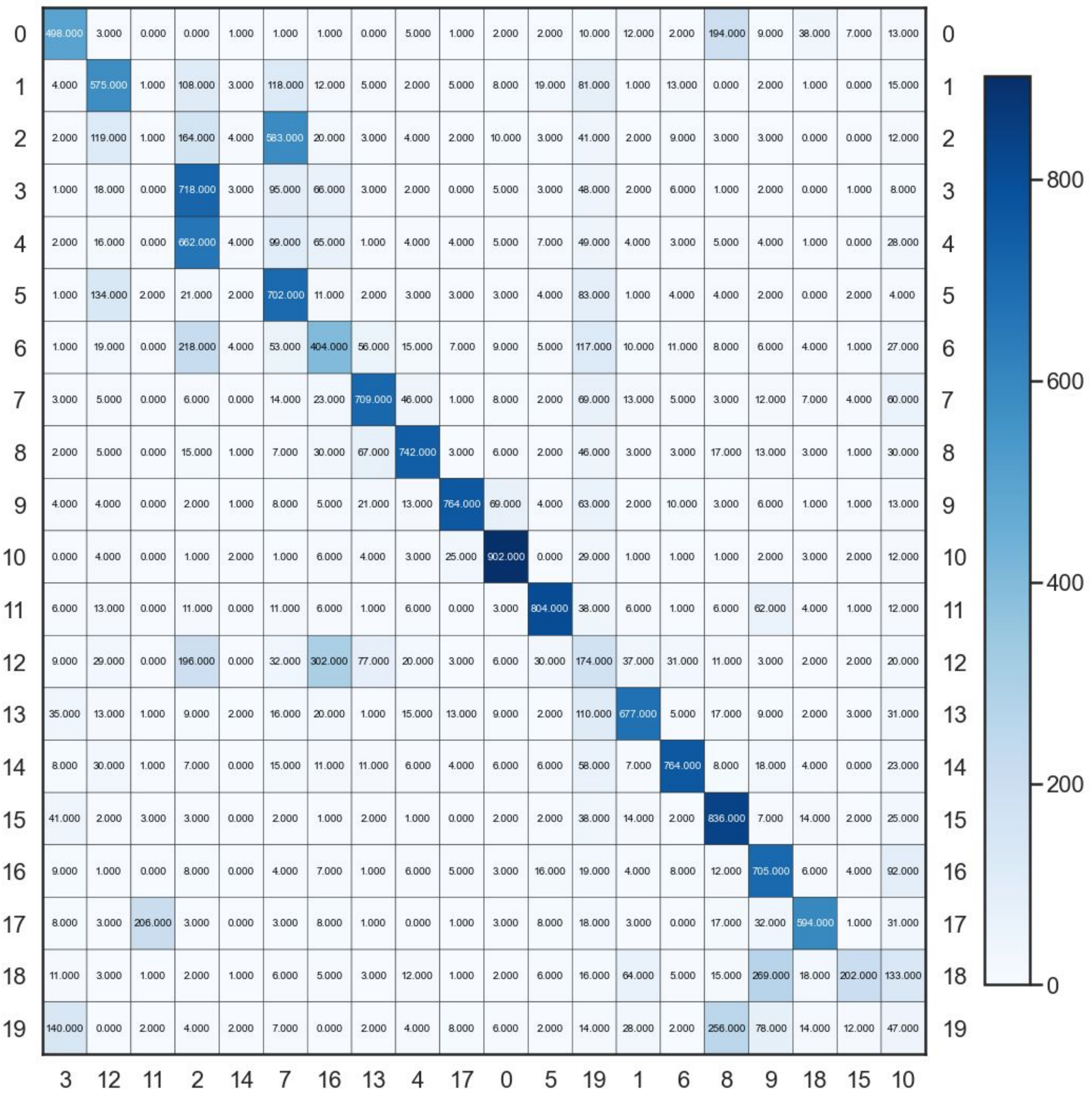
Completeness Score 0.5622405377922715

V-measure Score 0.5488326200602862

Adjusted Rand Index 0.41280179805176986

Adjusted Mutual Information Score 0.5473274512508833


Cosine
UMAP
r
0.1
0.2
0.3
0.4
0.5
0
25
50
75
100
125
150
175
200
homogeneity
completeness
v_measure
adjusted_rand_index
adjusted_mutual_info



	3	12	11	2	14	7	16	13	4	17	0	5	19	1	6	8	9	18	15	10
0	498.000	3.000	0.000	0.000	1.000	1.000	1.000	0.000	5.000	1.000	2.000	2.000	10.000	12.000	2.000	194.000	9.000	38.000	7.000	13.000
1	4.000	575.000	1.000	108.000	3.000	118.000	12.000	5.000	2.000	5.000	8.000	19.000	81.000	1.000	13.000	0.000	2.000	1.000	0.000	15.000
2	2.000	119.000	1.000	164.000	4.000	583.000	20.000	3.000	4.000	2.000	10.000	3.000	41.000	2.000	9.000	3.000	3.000	0.000	0.000	12.000
3	1.000	18.000	0.000	718.000	3.000	95.000	66.000	3.000	2.000	0.000	5.000	3.000	48.000	2.000	6.000	1.000	2.000	0.000	1.000	8.000
4	2.000	16.000	0.000	662.000	4.000	99.000	65.000	1.000	4.000	4.000	5.000	7.000	49.000	4.000	3.000	5.000	4.000	1.000	0.000	28.000
5	1.000	134.000	2.000	21.000	2.000	702.000	11.000	2.000	3.000	3.000	3.000	4.000	83.000	1.000	4.000	4.000	2.000	0.000	2.000	4.000
6	1.000	19.000	0.000	218.000	4.000	53.000	404.000	56.000	15.000	7.000	9.000	5.000	117.000	10.000	11.000	8.000	6.000	4.000	1.000	27.000
7	3.000	5.000	0.000	6.000	0.000	14.000	23.000	709.000	46.000	1.000	8.000	2.000	69.000	13.000	5.000	3.000	12.000	7.000	4.000	60.000
8	2.000	5.000	0.000	15.000	1.000	7.000	30.000	67.000	742.000	3.000	6.000	2.000	46.000	3.000	3.000	17.000	13.000	3.000	1.000	30.000
9	4.000	4.000	0.000	2.000	1.000	8.000	5.000	21.000	13.000	764.000	69.000	4.000	63.000	2.000	10.000	3.000	6.000	1.000	1.000	13.000
10	0.000	4.000	0.000	1.000	2.000	1.000	6.000	4.000	3.000	25.000	902.000	0.000	29.000	1.000	1.000	1.000	2.000	3.000	2.000	12.000
11	6.000	13.000	0.000	11.000	0.000	11.000	6.000	1.000	6.000	0.000	3.000	804.000	38.000	6.000	1.000	6.000	62.000	4.000	1.000	12.000
12	9.000	29.000	0.000	196.000	0.000	32.000	302.000	77.000	20.000	3.000	6.000	30.000	174.000	37.000	31.000	11.000	3.000	2.000	2.000	20.000
13	35.000	13.000	1.000	9.000	2.000	16.000	20.000	1.000	15.000	13.000	9.000	2.000	110.000	677.000	5.000	17.000	9.000	2.000	3.000	31.000
14	8.000	30.000	1.000	7.000	0.000	15.000	11.000	11.000	6.000	4.000	6.000	6.000	58.000	7.000	764.000	8.000	18.000	4.000	0.000	23.000
15	41.000	2.000	3.000	3.000	0.000	2.000	1.000	2.000	1.000	0.000	2.000	2.000	38.000	14.000	2.000	836.000	7.000	14.000	2.000	25.000
16	9.000	1.000	0.000	8.000	0.000	4.000	7.000	1.000	6.000	5.000	3.000	16.000	19.000	4.000	8.000	12.000	705.000	6.000	4.000	92.000
17	8.000	3.000	206.000	3.000	0.000	3.000	8.000	1.000	0.000	1.000	3.000	8.000	18.000	3.000	0.000	17.000	32.000	594.000	1.000	31.000
18	11.000	3.000	1.000	2.000	1.000	6.000	5.000	3.000	12.000	1.000	2.000	6.000	16.000	64.000	5.000	15.000	269.000	18.000	202.000	133.000
19	140.000	0.000	2.000	4.000	2.000	7.000	0.000	2.000	4.000	8.000	6.000	2.000	14.000	28.000	2.000	256.000	78.000	14.000	12.000	47.000

Question 12:

From the contingency matrices above, we can see that the Euclidean matrix is more prone to be confused while from the Cosine confusion matrix it is cleaner and less probable to be confused. The result matched our expectation since the Euclidean metric belongs to Minkowski style metrics and the Cosine metric belongs to angular and correction metrics. Those two set of metrics under UMAP instructions do have the behaviors we saw in our test cases.

Question 13:

For this question, we get following outputs:

“Ward”

Homogeneity Score 0.5017633893942085

Completeness Score 0.5266759659173146

V-measure Score 0.5139179406278006

Adjusted Rand Index 0.3760639944178731

Adjusted Mutual Information Score 0.5122957758731398

“single”

Homogeneity Score 0.020983070656432658

Completeness Score 0.3605274423617933

V-measure Score 0.0396580043722101

Adjusted Rand Index 0.0004612547204987331

Adjusted Mutual Information Score 0.034176765481306826

We can see that the linkage single does not work well, all inputs are clustered wrongly to zero.

Question 14:

For Dbscan, we tested metrics, min_samples and eps these three parameters. The best we get is with eps = 0.55, min_samples = 100 and metric is euclidean

min_sample: 100

Homogeneity 0.4666210225402587

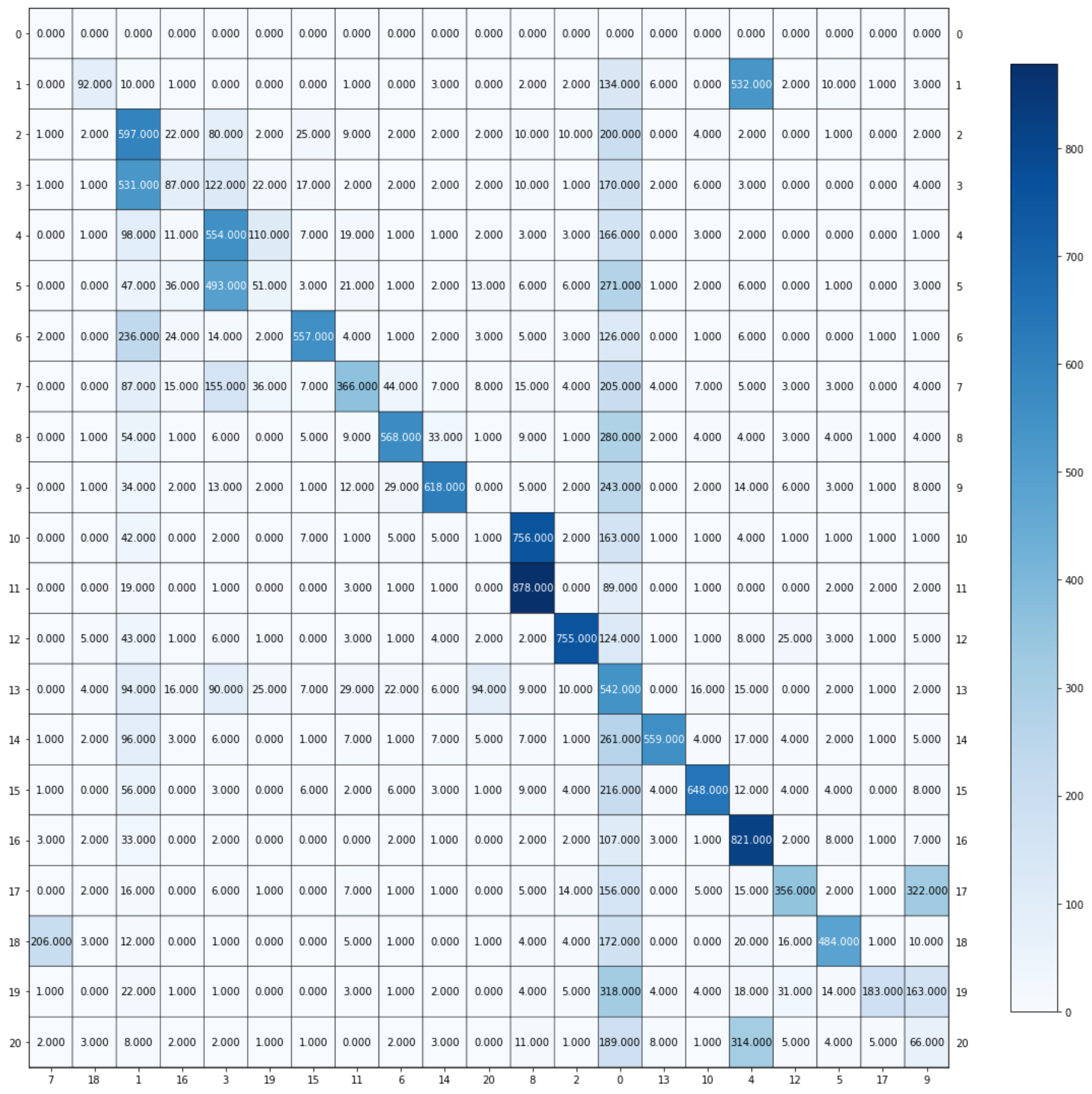
Completeness Score 0.5282599394438849

V-measure Score 0.4955310283931022

Adjusted Rand Index 0.2483640672034359

Adjusted Mutual Information Score 0.4936991642480833

metric is: euclidean



	7	18	1	16	3	19	15	11	6	14	20	8	2	0	13	10	4	12	5	17	9
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	92.000	10.000	1.000	0.000	0.000	0.000	1.000	0.000	3.000	0.000	2.000	2.000	134.000	6.000	0.000	532.000	2.000	10.000	1.000	3.000
2	1.000	2.000	597.000	22.000	80.000	2.000	25.000	9.000	2.000	2.000	2.000	10.000	10.000	200.000	0.000	4.000	2.000	0.000	1.000	0.000	2.000
3	1.000	1.000	531.000	87.000	122.000	22.000	17.000	2.000	2.000	2.000	2.000	10.000	1.000	170.000	2.000	6.000	3.000	0.000	0.000	0.000	4.000
4	0.000	1.000	98.000	11.000	554.000	110.000	7.000	19.000	1.000	1.000	2.000	3.000	3.000	166.000	0.000	3.000	2.000	0.000	0.000	0.000	1.000
5	0.000	0.000	47.000	36.000	493.000	51.000	3.000	21.000	1.000	2.000	13.000	6.000	6.000	271.000	1.000	2.000	6.000	0.000	1.000	0.000	3.000
6	2.000	0.000	236.000	24.000	14.000	2.000	557.000	4.000	1.000	2.000	3.000	5.000	3.000	126.000	0.000	1.000	6.000	0.000	0.000	1.000	1.000
7	0.000	0.000	87.000	15.000	155.000	36.000	7.000	366.000	44.000	7.000	8.000	15.000	4.000	205.000	4.000	7.000	5.000	3.000	3.000	0.000	4.000
8	0.000	1.000	54.000	1.000	6.000	0.000	5.000	9.000	568.000	33.000	1.000	9.000	1.000	280.000	2.000	4.000	4.000	3.000	4.000	1.000	4.000
9	0.000	1.000	34.000	2.000	13.000	2.000	1.000	12.000	29.000	618.000	0.000	5.000	2.000	243.000	0.000	2.000	14.000	6.000	3.000	1.000	8.000
10	0.000	0.000	42.000	0.000	2.000	0.000	7.000	1.000	5.000	5.000	1.000	756.000	2.000	163.000	1.000	1.000	4.000	1.000	1.000	1.000	1.000
11	0.000	0.000	19.000	0.000	1.000	0.000	0.000	3.000	1.000	1.000	0.000	878.000	0.000	89.000	0.000	1.000	0.000	0.000	2.000	2.000	2.000
12	0.000	5.000	43.000	1.000	6.000	1.000	0.000	3.000	1.000	4.000	2.000	2.000	755.000	124.000	1.000	1.000	8.000	25.000	3.000	1.000	5.000
13	0.000	4.000	94.000	16.000	90.000	25.000	7.000	29.000	22.000	6.000	94.000	9.000	10.000	542.000	0.000	16.000	15.000	0.000	2.000	1.000	2.000
14	1.000	2.000	96.000	3.000	6.000	0.000	1.000	7.000	1.000	7.000	5.000	7.000	1.000	261.000	559.000	4.000	17.000	4.000	2.000	1.000	5.000
15	1.000	0.000	56.000	0.000	3.000	0.000	6.000	2.000	6.000	3.000	1.000	9.000	4.000	216.000	4.000	648.000	12.000	4.000	4.000	0.000	8.000
16	3.000	2.000	33.000	0.000	2.000	0.000	0.000	0.000	2.000	1.000	0.000	2.000	2.000	107.000	3.000	1.000	821.000	2.000	8.000	1.000	7.000
17	0.000	2.000	16.000	0.000	6.000	1.000	0.000	7.000	1.000	1.000	0.000	5.000	14.000	156.000	0.000	5.000	15.000	356.000	2.000	1.000	322.000
18	206.000	3.000	12.000	0.000	1.000	0.000	0.000	5.000	1.000	0.000	1.000	4.000	4.000	172.000	0.000	0.000	20.000	16.000	484.000	1.000	10.000
19	1.000	0.000	22.000	1.000	1.000	0.000	0.000	3.000	1.000	2.000	0.000	4.000	5.000	318.000	4.000	4.000	18.000	31.000	14.000	183.000	163.000
20	2.000	3.000	8.000	2.000	2.000	1.000	1.000	0.000	2.000	3.000	0.000	11.000	1.000	189.000	8.000	1.000	314.000	5.000	4.000	5.000	66.000

For HDBSCAN we tested metrics, cluster_selection_epsilon and min_samples

eps: 0.2

min_sample: 150

Homogeneity Score 0.3750396494844003

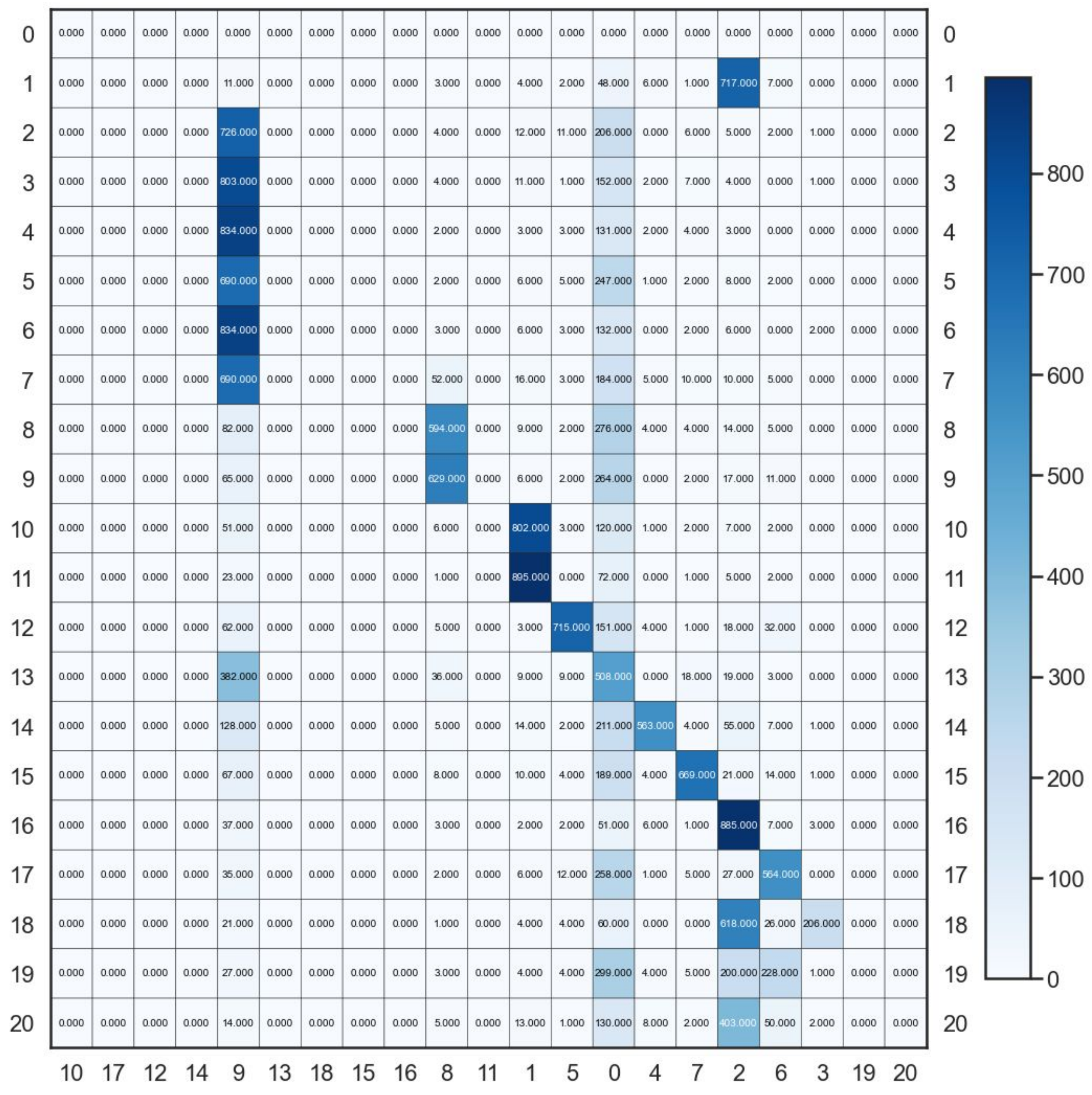
Completeness Score 0.5701629753929293

V-measure Score 0.45246112698448915

Adjusted Rand Index 0.19742645616795113

Adjusted Mutual Information Score 0.4514521062280271

metric is: canberra
)



	10	17	12	14	9	13	18	15	16	8	11	1	5	0	4	7	2	6	3	19	20
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	11.000	0.000	0.000	0.000	0.000	3.000	0.000	4.000	2.000	48.000	6.000	1.000	717.000	7.000	0.000	0.000	0.000
2	0.000	0.000	0.000	0.000	726.000	0.000	0.000	0.000	0.000	4.000	0.000	12.000	11.000	206.000	0.000	6.000	5.000	2.000	1.000	0.000	0.000
3	0.000	0.000	0.000	0.000	803.000	0.000	0.000	0.000	0.000	4.000	0.000	11.000	1.000	152.000	2.000	7.000	4.000	0.000	1.000	0.000	0.000
4	0.000	0.000	0.000	0.000	834.000	0.000	0.000	0.000	0.000	2.000	0.000	3.000	3.000	131.000	2.000	4.000	3.000	0.000	0.000	0.000	0.000
5	0.000	0.000	0.000	0.000	690.000	0.000	0.000	0.000	0.000	2.000	0.000	6.000	5.000	247.000	1.000	2.000	8.000	2.000	0.000	0.000	0.000
6	0.000	0.000	0.000	0.000	834.000	0.000	0.000	0.000	0.000	3.000	0.000	6.000	3.000	132.000	0.000	2.000	6.000	0.000	2.000	0.000	0.000
7	0.000	0.000	0.000	0.000	690.000	0.000	0.000	0.000	0.000	52.000	0.000	16.000	3.000	184.000	5.000	10.000	10.000	5.000	0.000	0.000	0.000
8	0.000	0.000	0.000	0.000	82.000	0.000	0.000	0.000	0.000	594.000	0.000	9.000	2.000	276.000	4.000	4.000	14.000	5.000	0.000	0.000	0.000
9	0.000	0.000	0.000	0.000	65.000	0.000	0.000	0.000	0.000	629.000	0.000	6.000	2.000	264.000	0.000	2.000	17.000	11.000	0.000	0.000	0.000
10	0.000	0.000	0.000	0.000	51.000	0.000	0.000	0.000	0.000	6.000	0.000	802.000	3.000	120.000	1.000	2.000	7.000	2.000	0.000	0.000	0.000
11	0.000	0.000	0.000	0.000	23.000	0.000	0.000	0.000	0.000	1.000	0.000	895.000	0.000	72.000	0.000	1.000	5.000	2.000	0.000	0.000	0.000
12	0.000	0.000	0.000	0.000	62.000	0.000	0.000	0.000	0.000	5.000	0.000	3.000	715.000	151.000	4.000	1.000	18.000	32.000	0.000	0.000	0.000
13	0.000	0.000	0.000	0.000	382.000	0.000	0.000	0.000	0.000	36.000	0.000	9.000	9.000	508.000	0.000	18.000	19.000	3.000	0.000	0.000	0.000
14	0.000	0.000	0.000	0.000	128.000	0.000	0.000	0.000	0.000	5.000	0.000	14.000	2.000	211.000	563.000	4.000	55.000	7.000	1.000	0.000	0.000
15	0.000	0.000	0.000	0.000	67.000	0.000	0.000	0.000	0.000	8.000	0.000	10.000	4.000	189.000	4.000	669.000	21.000	14.000	1.000	0.000	0.000
16	0.000	0.000	0.000	0.000	37.000	0.000	0.000	0.000	0.000	3.000	0.000	2.000	2.000	51.000	6.000	1.000	885.000	7.000	3.000	0.000	0.000
17	0.000	0.000	0.000	0.000	35.000	0.000	0.000	0.000	0.000	2.000	0.000	6.000	12.000	258.000	1.000	5.000	27.000	564.000	0.000	0.000	0.000
18	0.000	0.000	0.000	0.000	21.000	0.000	0.000	0.000	0.000	1.000	0.000	4.000	4.000	60.000	0.000	0.000	618.000	26.000	206.000	0.000	0.000
19	0.000	0.000	0.000	0.000	27.000	0.000	0.000	0.000	0.000	3.000	0.000	4.000	4.000	299.000	4.000	5.000	200.000	228.000	1.000	0.000	0.000
20	0.000	0.000	0.000	0.000	14.000	0.000	0.000	0.000	0.000	5.000	0.000	13.000	1.000	130.000	8.000	2.000	403.000	50.000	2.000	0.000	0.000

Question 15:

There are 21 clusters in the DBSCAN model and 10 clusters in HDBSCAN model. HDBSCAN is a noise aware cluster algorithm. The -1 label are the data samples that are not assigned to any clusters, -1 results are outliers,

Question 16:

1. For this part, I downloaded the datasets on my PC and used the pandas.read_csv function to store the data in a pandas dataframe.

2. Then I changed the ‘Category’ column into categorical objects and created a new column named ‘Category ID’ which corresponds to the 5 categories. 0 is business, 1 is entertainment, 2 is politics, 3 is sport and 4 is tech.
3. The next step is to transform the Text column into TF-IDF vectors.
4. First, I tried to use SVD to reduce data and use KMeans() clustering. I tested several r from 1 to 300. However, the best V-measure score I got is only around 0.235.
5. And I decided to use the UMAP with parameters n_components = 100, metric = 'cosine' to reduce the dimensionality of the TF-IDF matrix. Then use Kmeans clustering method with parameters KMeans(n_clusters=5, random_state=0, max_iter=1000, n_init=30), we reached a very decent result.

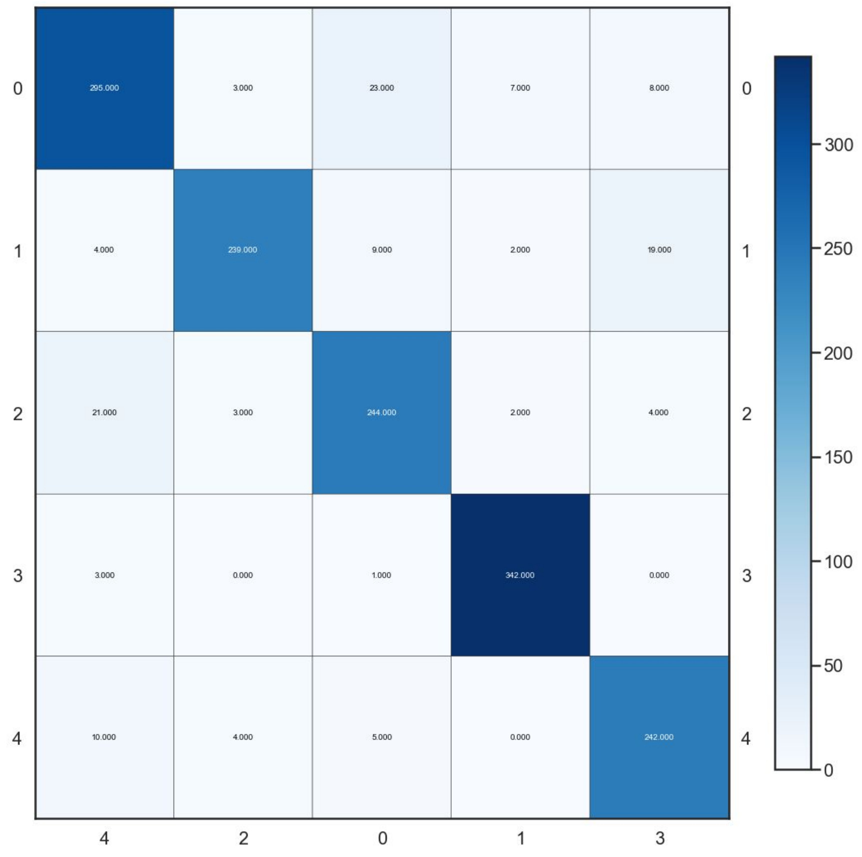
   Homogeneity Score 0.7708326965955365

   Completeness Score 0.7715061495302332

   V-measure Score 0.7711692760333082

   Adjusted Rand Index 0.8046862093949397

   Adjusted Mutual Information Score 0.7703963769832721



0
1
2
3
4
295.000
3.000
23.000
7.000
8.000
4.000
239.000
9.000
2.000
19.000
21.000
3.000
244.000
2.000
4.000
3.000
0.000
1.000
342.000
0.000
10.000
4.000
5.000
0.000
242.000
4
2
0
1
3
300
250
200
150
100
50
0