# EE219 Project 2 Report

## Clustering

## Winter 2018

**Will Li (004356402)**
**Ruiyi Wu (304615036)**

1. **Building the TF-IDF matrix.**

After excluding the stop words, we obtained the dimensions of TF-IDF matrix for min_df = 3 as:

```
For min_df = 3, TFxIDF matrix: (7882, 27768)
```

2. **K-mean clustering with k = 2 using TF-IDF data**

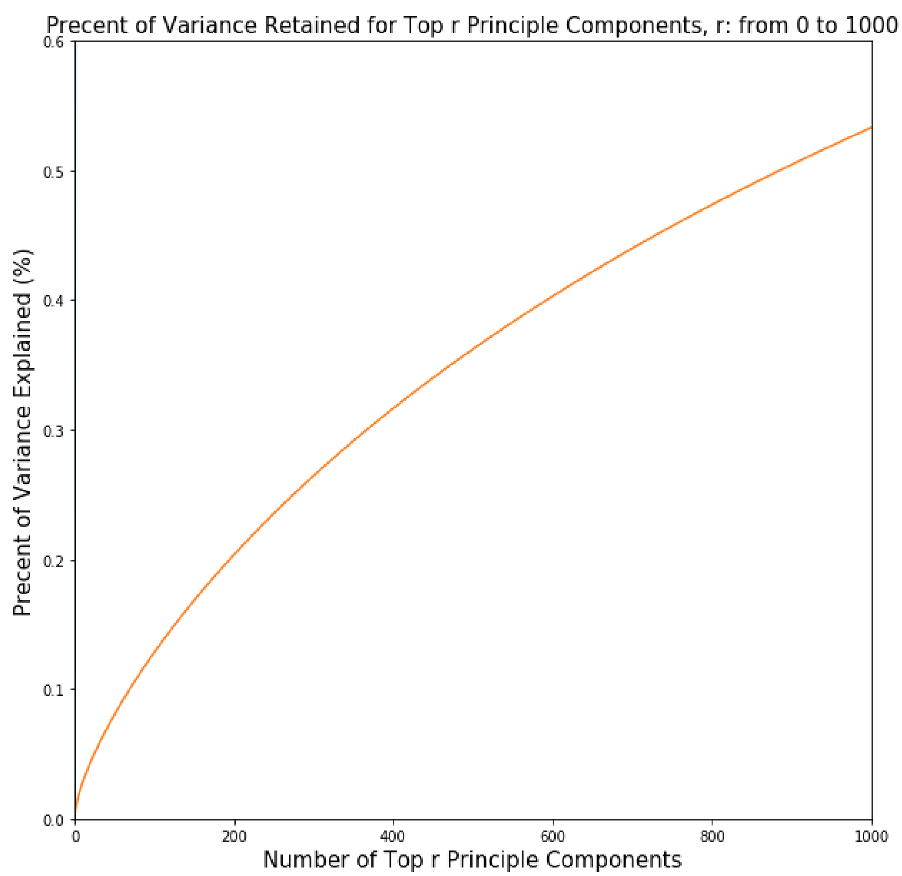
After applying the five required measures, we obtained the following K-mean clustering results:

```
contingency matrix:
[[3899    4]
 [2262 1717]]
Homogeneity Score: 0.253
Completeness Score: 0.335
V-measure: 0.288
Adjusted Rand Score: 0.181
Adjusted Mutual Info Score: 0.253
```

As we can see from the accuracy for each measure that high dimensional TF-IDF vectors do not yield good results.

3. **Preprocess the data before clustering**

a) Using the method of explained variance ratio, we obtained the percent of variance the top r (=1000) principle components can retain is 0.534503861455.



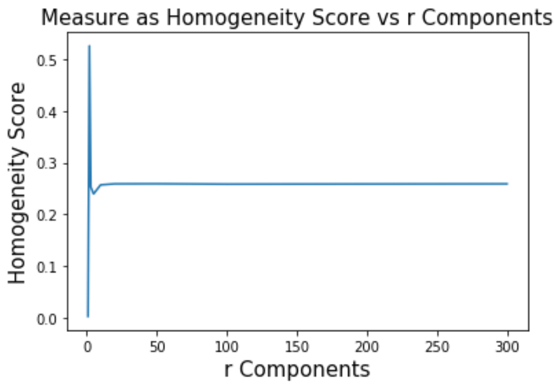
b)

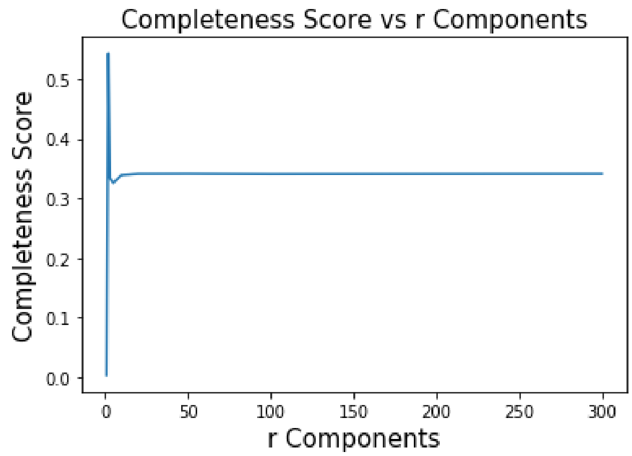
- LSI:

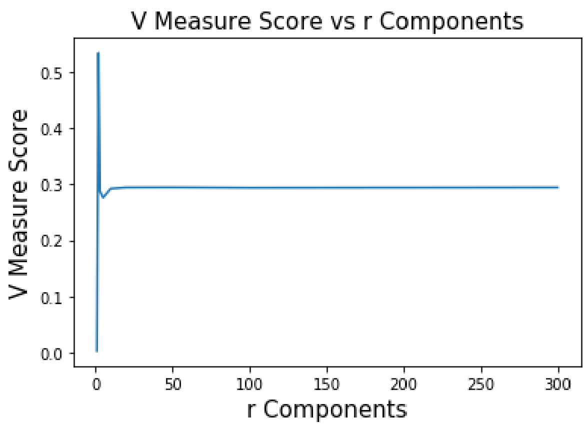
```
Number of components: 1
contingency matrix:
[[1739 2164]
 [1508 2471]]
Homogeneity Score: 0.003
Completeness Score: 0.003
V-measure: 0.003
Adjusted Rand Score: 0.005
Adjusted Mutual Info Score: 0.003
=================================================
Number of components: 2
contingency matrix:
[[3852   51]
 [ 916 3063]]
Homogeneity Score: 0.525
Completeness Score: 0.543
V-measure: 0.534
Adjusted Rand Score: 0.569
Adjusted Mutual Info Score: 0.525
=================================================
Number of components: 3
contingency matrix:
[[3898    5]
 [2259 1720]]
Homogeneity Score: 0.253
Completeness Score: 0.334
V-measure: 0.288
Adjusted Rand Score: 0.181
Adjusted Mutual Info Score: 0.253
=================================================
```

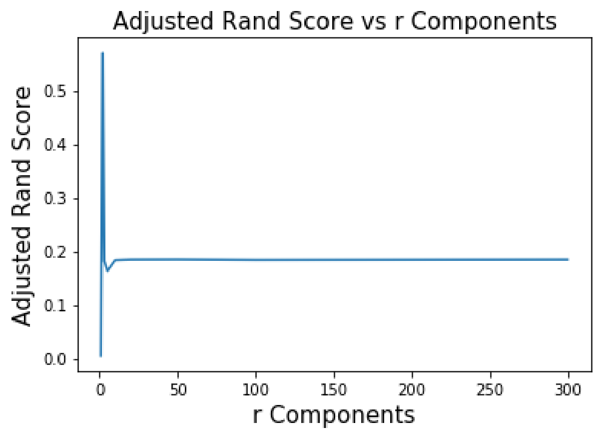
```
Number of components: 5
contingency matrix:
[[   2 3901]
 [1629 2350]]
Homogeneity Score: 0.240
Completeness Score: 0.326
V-measure: 0.276
Adjusted Rand Score: 0.162
Adjusted Mutual Info Score: 0.240
=================================================
Number of components: 10
contingency matrix:
[[3901    2]
 [2254 1725]]
Homogeneity Score: 0.257
Completeness Score: 0.339
V-measure: 0.292
Adjusted Rand Score: 0.183
Adjusted Mutual Info Score: 0.257
=================================================
Number of components: 20
contingency matrix:
[[   1 3902]
 [1729 2250]]
Homogeneity Score: 0.259
Completeness Score: 0.341
V-measure: 0.294
Adjusted Rand Score: 0.184
Adjusted Mutual Info Score: 0.259
=================================================
```

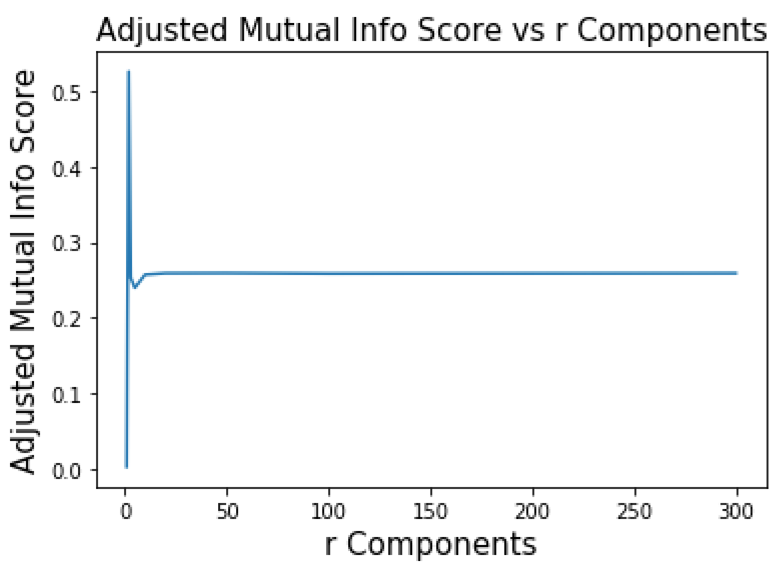
```
Number of components: 50
contingency matrix:
[[   1 3902]
 [1730 2249]]
Homogeneity Score: 0.259
Completeness Score: 0.341
V-measure: 0.295
Adjusted Rand Score: 0.184
Adjusted Mutual Info Score: 0.259
=================================================
Number of components: 100
contingency matrix:
[[3902    1]
 [2253 1726]]
Homogeneity Score: 0.258
Completeness Score: 0.341
V-measure: 0.294
Adjusted Rand Score: 0.183
Adjusted Mutual Info Score: 0.258
=================================================
Number of components: 300
contingency matrix:
[[3902    1]
 [2250 1729]]
Homogeneity Score: 0.259
Completeness Score: 0.341
V-measure: 0.294
Adjusted Rand Score: 0.184
Adjusted Mutual Info Score: 0.259
=================================================
```



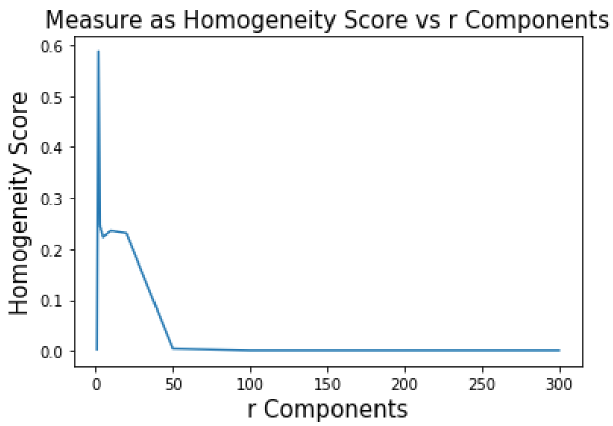



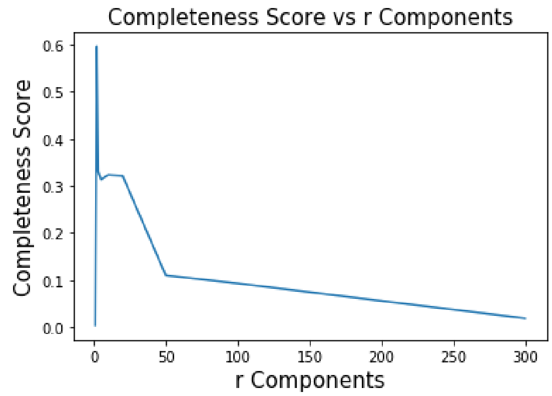



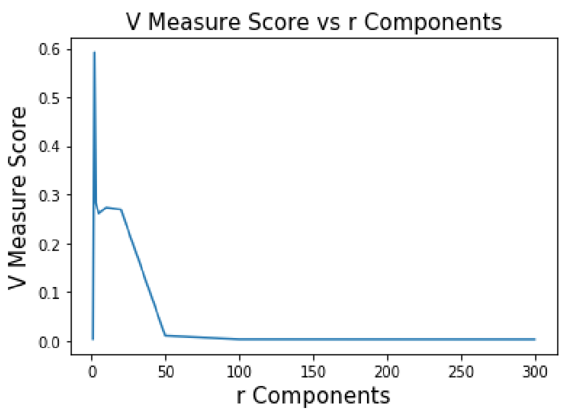

As we can see from the accuracy of the measures and plots that with the number of components equal to **2** , the LSI reduction yield best results among all the dimension parameters.

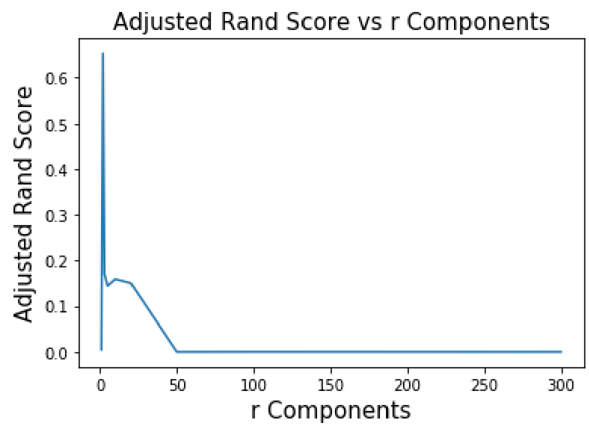
- NMF:

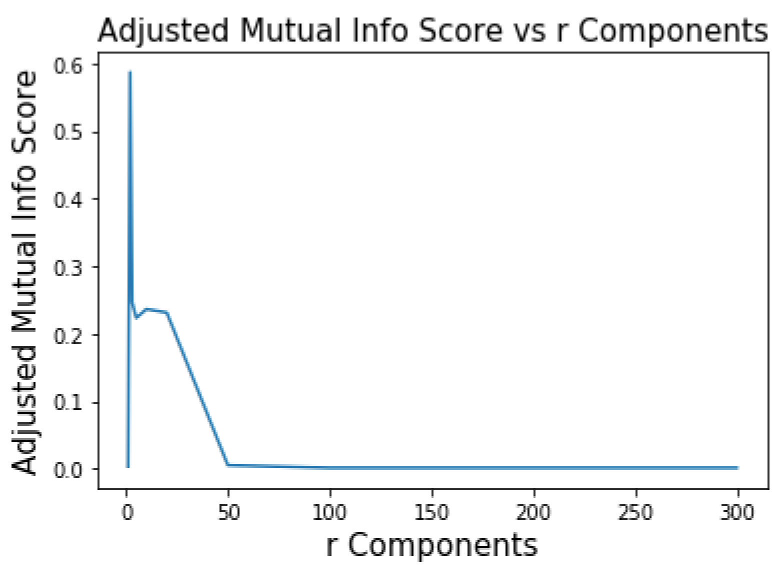
```
Number of components: 1
Homogeneity Score: 0.003
Completeness Score: 0.003
V-measure: 0.003
Adjusted Rand Score: 0.005
Adjusted Mutual Info Score: 0.003
contingency matrix:
[[1507 2472]
 [1739 2164]]
================================================
Number of components: 2
Homogeneity Score: 0.587
Completeness Score: 0.597
V-measure: 0.592
Adjusted Rand Score: 0.654
Adjusted Mutual Info Score: 0.587
contingency matrix:
[[ 689 3290]
 [3837   66]]
================================================
Number of components: 3
Homogeneity Score: 0.247
Completeness Score: 0.331
V-measure: 0.283
Adjusted Rand Score: 0.171
Adjusted Mutual Info Score: 0.247
contingency matrix:
[[1668 2311]
 [   2 3901]]
================================================
Number of components: 5
Homogeneity Score: 0.223
Completeness Score: 0.314
V-measure: 0.261
Adjusted Rand Score: 0.144
Adjusted Mutual Info Score: 0.223
contingency matrix:
[[2442 1537]
 [3901    2]]
================================================
Number of components: 10
Homogeneity Score: 0.237
Completeness Score: 0.323
V-measure: 0.273
Adjusted Rand Score: 0.159
Adjusted Mutual Info Score: 0.237
contingency matrix:
[[2367 1612]
 [3901    2]]
================================================
Number of components: 20
Homogeneity Score: 0.231
Completeness Score: 0.322
V-measure: 0.269
Adjusted Rand Score: 0.151
Adjusted Mutual Info Score: 0.231
contingency matrix:
[[2411 1568]
 [3903    0]]
================================================

Number of components: 50
Homogeneity Score: 0.005
Completeness Score: 0.110
V-measure: 0.010
Adjusted Rand Score: -0.000
Adjusted Mutual Info Score: 0.005
contingency matrix:
[[3937   42]
 [3903    0]]
================================================
Number of components: 100
Homogeneity Score: 0.001
Completeness Score: 0.093
V-measure: 0.003
Adjusted Rand Score: 0.000
Adjusted Mutual Info Score: 0.001
contingency matrix:
[[3979    0]
 [3892   11]]
================================================
Number of components: 300
Homogeneity Score: 0.001
Completeness Score: 0.019
V-measure: 0.003
Adjusted Rand Score: -0.000
Adjusted Mutual Info Score: 0.001
contingency matrix:
[[3925   54]
 [3883   20]]
================================================
```




For this section, as mentioned in Piazza (https://piazza.com/class/jcifzza0hzs2f3?cid=102), we used the explained_variance_ratio_ method to calculate the percentage of variance accounted. As we can see from the accuracy of the measures and plots that with the number of components as **2** will the NMF reduction yield best results among all the dimension parameters.

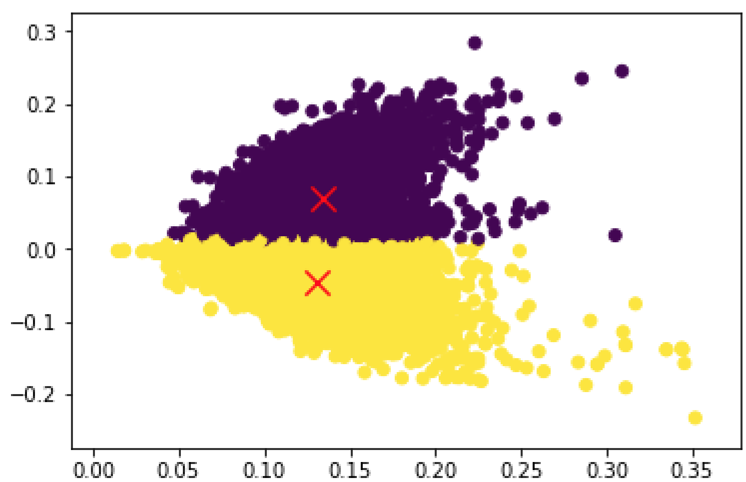
**Question**: How do you explain the non-monotonic behavior of the measures as r increases?

For SVD, we note that the peak at a low dimension value occurs at r=3. NMF has a very obvious peak at r=3, and similarly to SVD. It is also non-monotonic with a dip occuring at r=5, 10, recovering a very small amount at r=20, and then a further dip at r=50 and plateauing out beyond that point. We believe the graphs are non-monotonic because of opposing factors -- while more information is maintained by the TFxIDF matrix at high dimensions, K-Means clustering is not particularly effective at high dimensions. This has been explained in the project description. Also, different dimension might have different property which will also cause the non-monotonic behaviour.

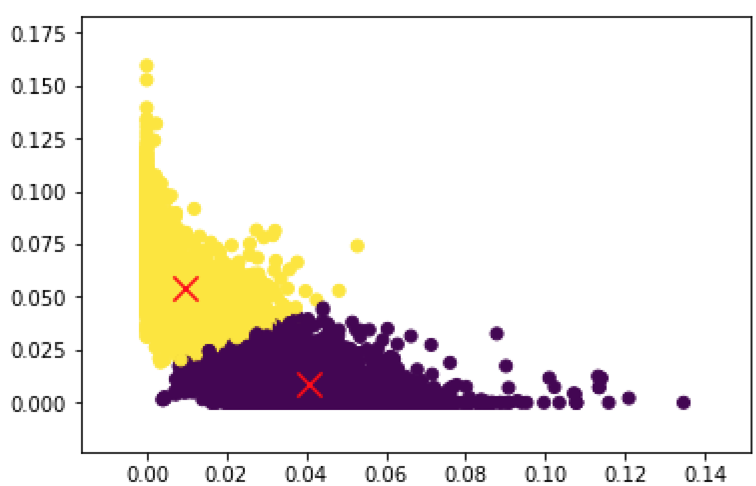
4. **Visualization**

a. 2D projection of final data vectors and color coding, with best r for clustering results

- LSI with best r = 2



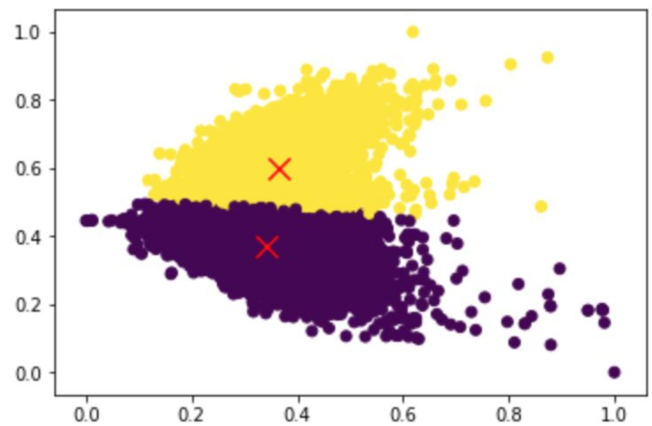

- NMF with best r = 2



b. Three methods that could increase performance; for each: repeat part a, report new measures

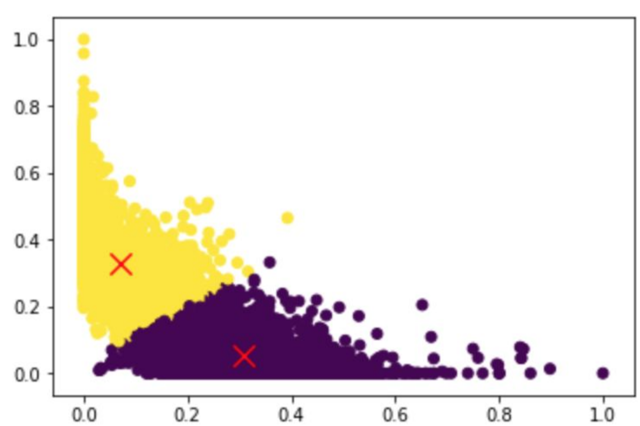
i. Feature Normalization

- LSI:



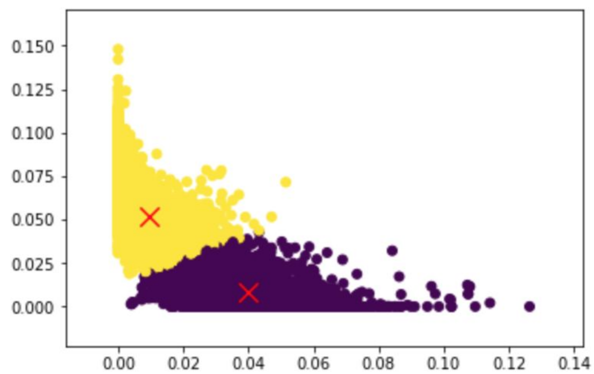
```
Contingency Matrix for LSI Feature Normalization:
[[3874   29]
 [1222 2757]]
Homogeneity: 0.457
Completeness: 0.487
V-measure: 0.471
Adjusted Rand-Index: 0.466
Adjusted Mutual info score: 0.457
```

- NMF:



```
Contingency Matrix for NMF Feature Normalization:
[[3824   79]
 [ 537 3442]]
Homogeneity: 0.633
Completeness: 0.638
V-measure: 0.636
Adjusted Rand-Index: 0.712
Adjusted Mutual info score: 0.633
```

ii. Logarithm Transformation to data vectors only after NMF



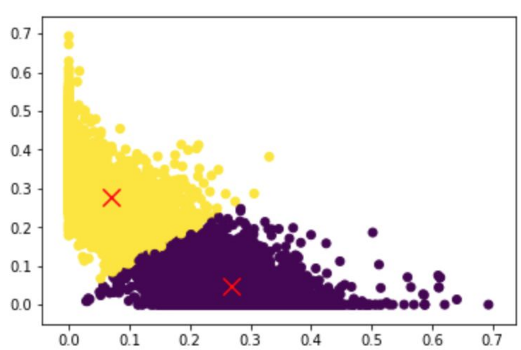

```
Contingency Matrix for Logarithm Non-linear Transformation:
[[3833   70]
 [ 649 3330]]
Homogeneity: 0.598
Completeness: 0.606
V-measure: 0.602
Adjusted Rand-Index: 0.668
Adjusted Mutual info score: 0.598
```

**Question:** Can you justify why logarithm transformation may increase the clustering results?

The main reason is that the log transformation can decrease the variability of data and make data conform more closely to the normal distribution. It reduced the variance of data set and make the same data more concentrate to one point. In this way, logarithm transformation may increase the clustering results.

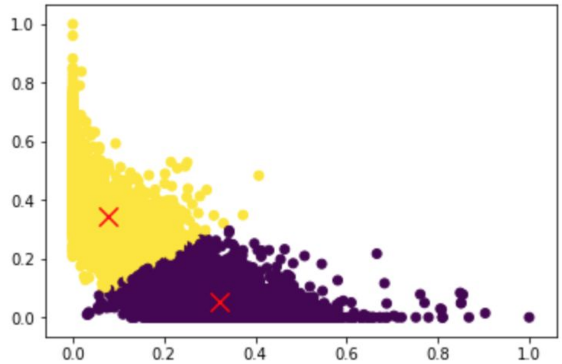
iii. Combining both transformations (in different orders) on NMF- reduced data.

- Normalization + Logarithm




```
Contingency Matrix for Combined Transformation Nor + Log :
[[3803  100]
 [ 475 3504]]
Homogeneity: 0.643
Completeness: 0.647
V-measure: 0.645
Adjusted Rand-Index: 0.729
Adjusted Mutual info score: 0.643
```

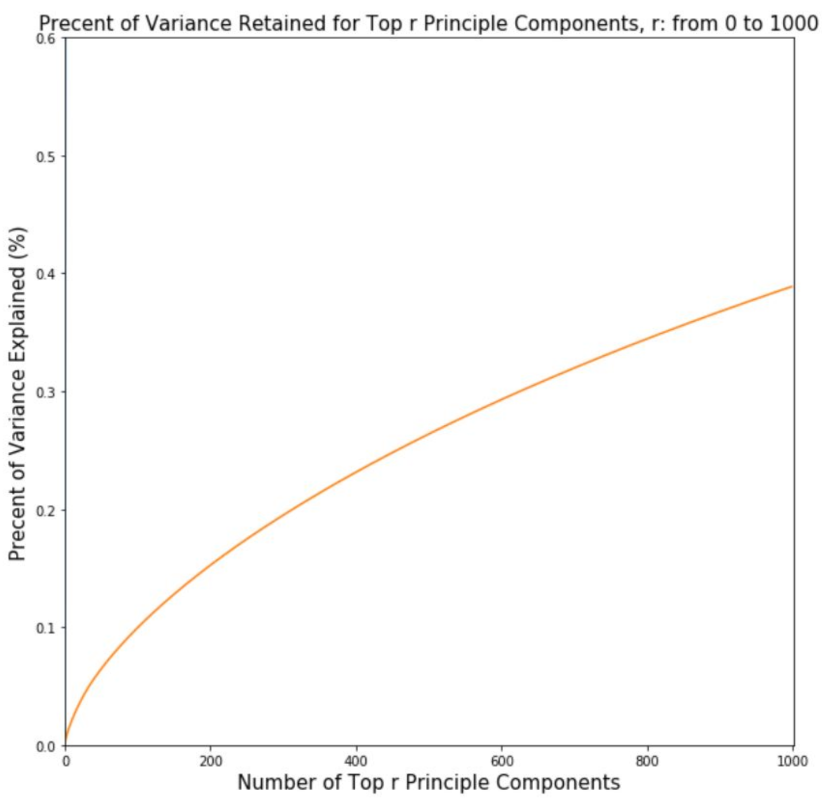
- <u>Logarithm+Normalization</u>




```
Contingency Matrix for Combined Transformation Log + Nor :
[[3817   86]
 [ 517 3462]]
Homogeneity: 0.636
Completeness: 0.640
V-measure: 0.638
Adjusted Rand-Index: 0.717
Adjusted Mutual info score: 0.636
```

## 5. Expand Dataset into 20 categories

**a.** For min_df = 3, TFxIDF matrix: (18846, 52295)



**b.** Example of Confusion matrix:

```
contingency matrix:
[[  1 193 146   0   0   0   1   0   0  26 268  11  34  32  15   2   0   0
    0  70]
 [ 18 341   0   0 387   0   0   1  79  20   0   9   6  35   2   0   0   3
    0  72]
 [  3 168   0   0 600   0   0   9 121  19   0  15   1  11   6   0   0   2
    0  30]
 [ 17 200   0   0  75   1   0   6 558   9   0  34   0  37   6   0   0  14
    0  25]
 [ 13 338   0   0  26   0   0   1 449  24   0  30   2  23  20   0   0  13
    0  24]
 [ 32 182   0   0 654   0   0   2  11   3   0   5   0   9  20   0   0   1
    0  69]
 [  3 181   0   0   8   3   0   3  52   5   0  43   0  26  14   0   0 632
    0   5]
 [ 14 716   0   0   2   0   0   1   0  50   0  53  48  50  15   1   0  14
    0  26]
 [ 22 718   0   0   0   0   0   1   0  12   0  17  17  69  11   0   0  17
    0 112]
 [  6 353   0   0   1 536   0   1   0  32   0   8   3  43   6   0   0   3
    0   2]
 [  4 123   0   0   0 751   0   0   0   2   0  12   1  47  56   0   0   1
    0   2]
 [  3 175   0   0  29   0 635  29   2   5   0   7  38   9  10   0   0   0
    0  49]
 [ 36 718   2   0  34   1   0  13  62  14   0   7   3  29   7   0   0   8
    0  50]
 [ 28 688   0   0   5   0   0   3   0  10   3  10 107  23  11   0   0   1
   78  23]
 [207 398   0   0   4   0   0 241   1  47   0   2  37   9  14   0   0   2
    0  25]
 [  5 215   0   0   2   0   0   0   0  12 703   3  17   4  15   3   0   0
    0  18]
 [  6 153   0   0   1   0   5   1   0  29   2  13 332  14   5 335   0   5
    0   9]
 [  0 137   0 415   0   0   0   0   0  18   4  18 109  36   3   0 197   0
    0   3]
 [ 25 209   0   0   0   2   2   1   0  23   7  52 228  26   0  79   0   0
    0 121]
 [  0 177  17   0   0   0   0   4   0  10 252  22  20  35   7  62   0   0
    2 201]]
```

Homogeneity Score: 0.375

Completeness Score: 0.462

V-measure: 0.414

Adjusted Rand Score: 0.145

Adjusted Mutual Info Score: 0.373

Note: We didn't include all the confusion matrix in the report. The matrix is 20* 20 which is huge and it doesn't have a good format in the Jupyter notebook.

**Discussion**: The result has agree with the case where  k = 2, that the classification is not very satisfied before the dimension reduction.

**c.**

<u>SVD:</u>

**Number of components: 1**

Homogeneity Score: 0.022

Completeness Score: 0.023

V-measure: 0.022

Adjusted Rand Score: 0.004

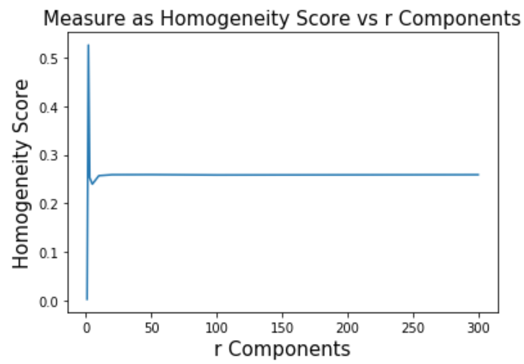
Adjusted Mutual Info Score: 0.018

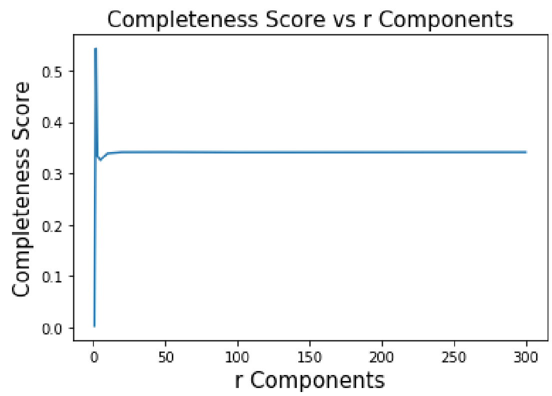
**Number of components: 2**

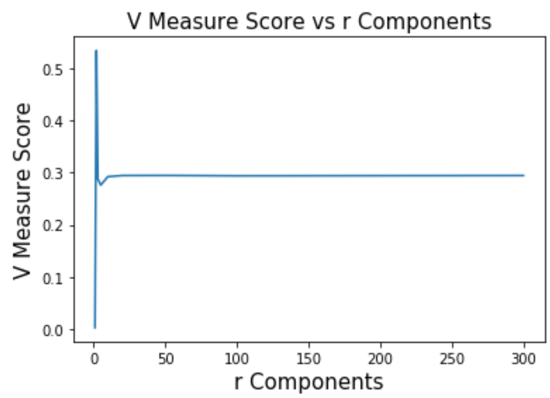
Homogeneity Score: 0.198
Completeness Score: 0.209
V-measure: 0.203
Adjusted Rand Score: 0.061
Adjusted Mutual Info Score: 0.195
**Number of components: 3**
Homogeneity Score: 0.260
Completeness Score: 0.271
V-measure: 0.266
Adjusted Rand Score: 0.092
Adjusted Mutual Info Score: 0.258
**Number of components: 5**
Homogeneity Score: 0.332
Completeness Score: 0.351
V-measure: 0.341
Adjusted Rand Score: 0.136
Adjusted Mutual Info Score: 0.330
**Number of components: 10**
Homogeneity Score: 0.367
Completeness Score: 0.403
V-measure: 0.384
Adjusted Rand Score: 0.169
Adjusted Mutual Info Score: 0.365
**Number of components: 20**
Homogeneity Score: 0.359
Completeness Score: 0.417
V-measure: 0.386
Adjusted Rand Score: 0.166
Adjusted Mutual Info Score: 0.357
**Number of components: 50**
Homogeneity Score: 0.360
Completeness Score: 0.467
V-measure: 0.407
Adjusted Rand Score: 0.154
Adjusted Mutual Info Score: 0.358
**Number of components: 100**
Homogeneity Score: 0.399
Completeness Score: 0.492
V-measure: 0.441
Adjusted Rand Score: 0.187
Adjusted Mutual Info Score: 0.397
**Number of components: 300**
Homogeneity Score: 0.348
Completeness Score: 0.445

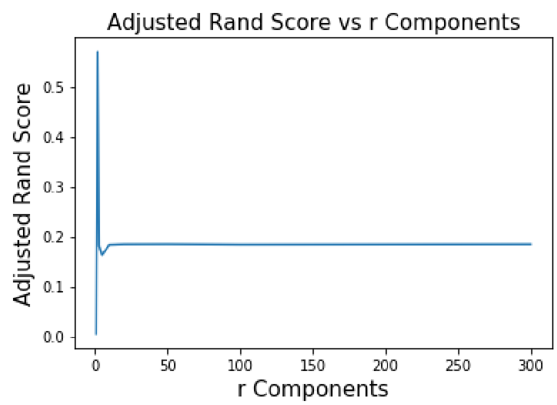
V-measure: 0.390
Adjusted Rand Score: 0.159
Adjusted Mutual Info Score: 0.345

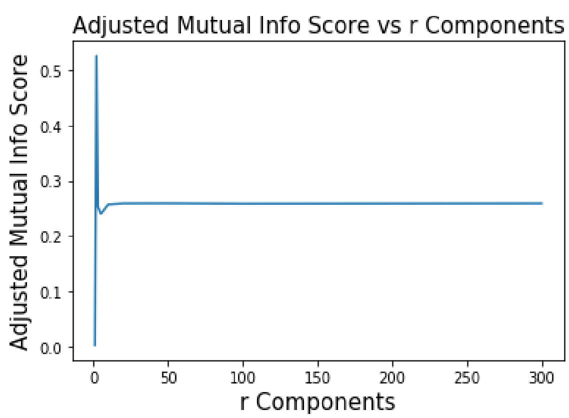
**Plot:**



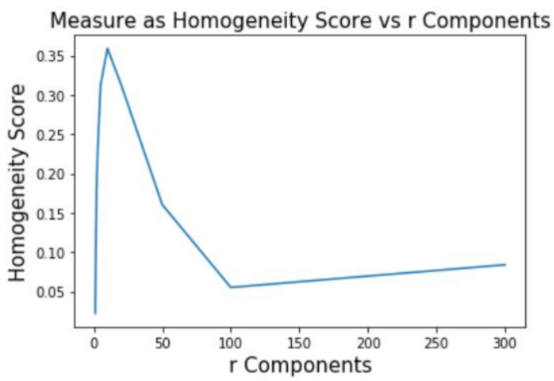

**Discussion**: In the case of 20 clustering for SVD, the result is still similar to the case where K = 2. The only difference is the average result is worst since 20 clustering is more probably to have misclassification. As mentioned in the previous part,  the result is maximum at r = 2 or 3 (low dimension).

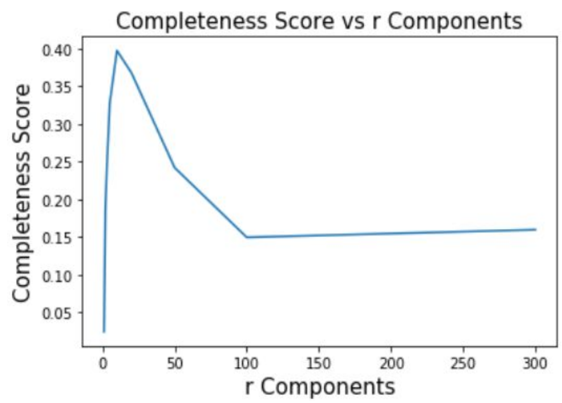
NMF:

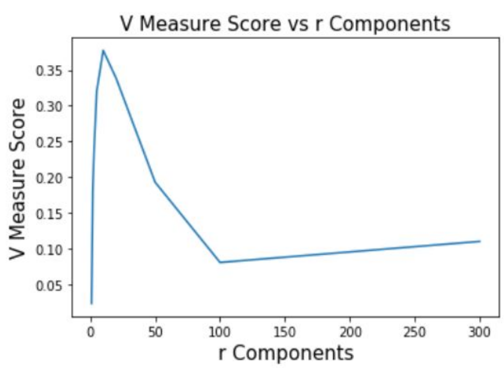
**Number of components: 1**
Homogeneity Score: 0.022
Completeness Score: 0.024
V-measure: 0.023
Adjusted Rand Score: 0.004
Adjusted Mutual Info Score: 0.019
**Number of components: 2**
Homogeneity Score: 0.179
Completeness Score: 0.187
V-measure: 0.183
Adjusted Rand Score: 0.057
Adjusted Mutual Info Score: 0.176
**Number of components: 3**
contingency matrix:
Homogeneity Score: 0.236
Completeness Score: 0.243
V-measure: 0.240
Adjusted Rand Score: 0.082
Adjusted Mutual Info Score: 0.234
**Number of components: 5**
Homogeneity Score: 0.313
Completeness Score: 0.328
V-measure: 0.320
Adjusted Rand Score: 0.120
Adjusted Mutual Info Score: 0.311
**Number of components: 10**
Homogeneity Score: 0.359
Completeness Score: 0.398
V-measure: 0.377
Adjusted Rand Score: 0.160
Adjusted Mutual Info Score: 0.357
**Number of components: 20**
Homogeneity Score: 0.313
Completeness Score: 0.368
V-measure: 0.338
Adjusted Rand Score: 0.134
Adjusted Mutual Info Score: 0.311
**Number of components: 50**
Homogeneity Score: 0.160
Completeness Score: 0.242
V-measure: 0.193
Adjusted Rand Score: 0.043
Adjusted Mutual Info Score: 0.158
**Number of components: 100**

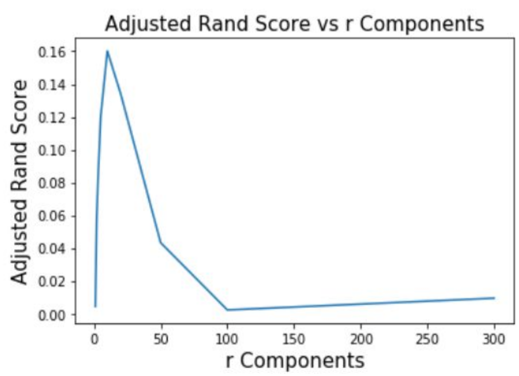
Homogeneity Score: 0.055
Completeness Score: 0.150
V-measure: 0.081
Adjusted Rand Score: 0.002
Adjusted Mutual Info Score: 0.052
**Number of components: 300**
Homogeneity Score: 0.084
Completeness Score: 0.160
V-measure: 0.110
Adjusted Rand Score: 0.009
Adjusted Mutual Info Score: 0.081

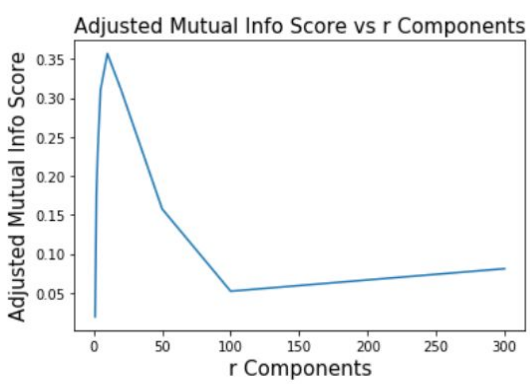
**Plot:**



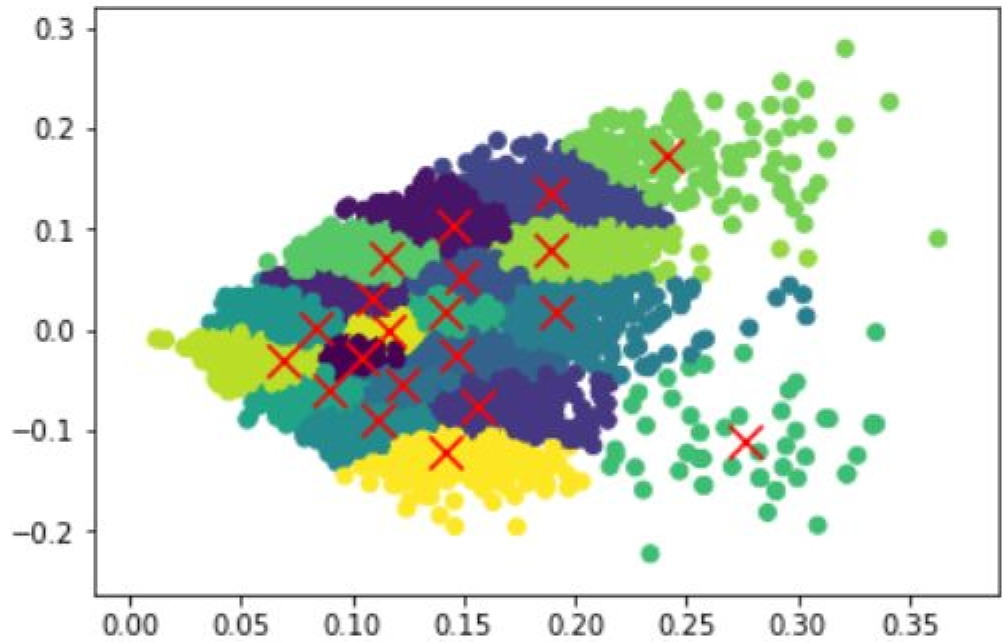



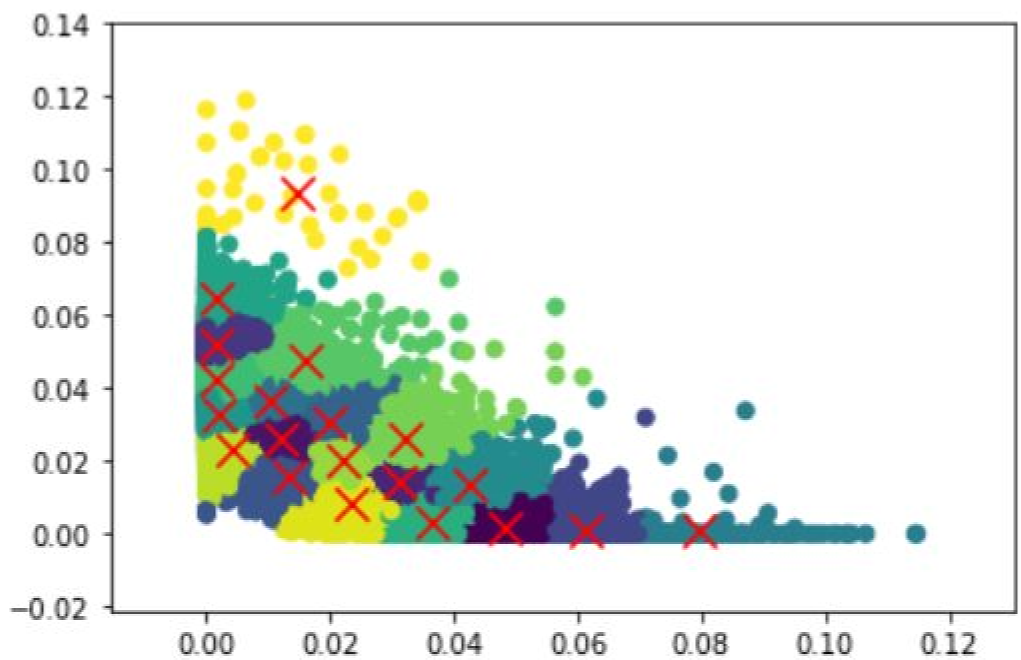
**Discussion:**
Again, the result is similar to the case when k = 2. The best result still occur around r equal to 2 or 3. For the last part or part 5, we will us r=2  for all the following case.

**d.** Visual Plot:

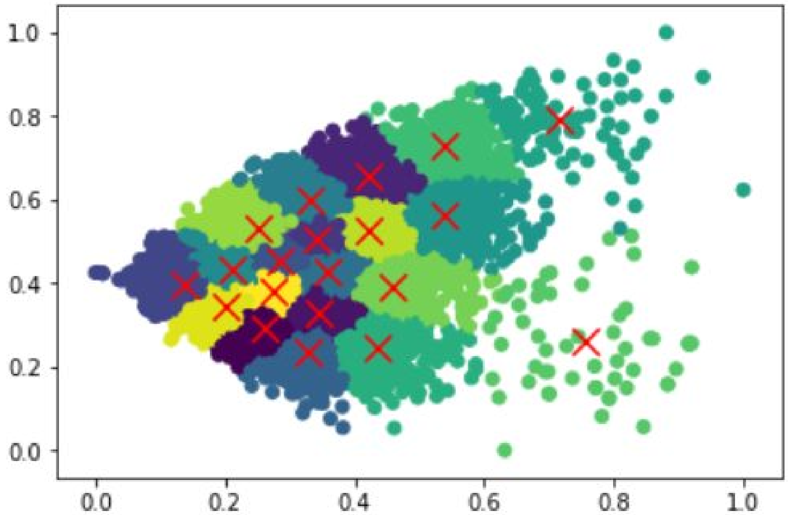
1. LSI




2. NMF

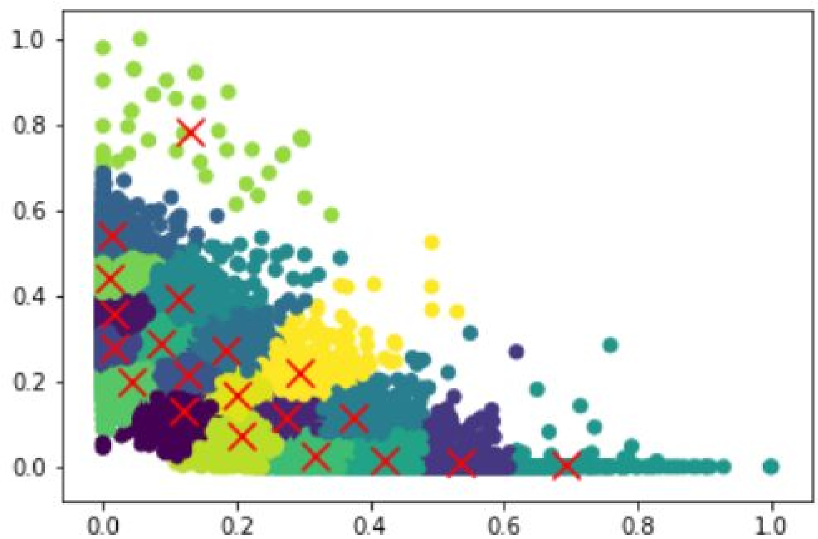


3. <u>Normalized LSI</u>



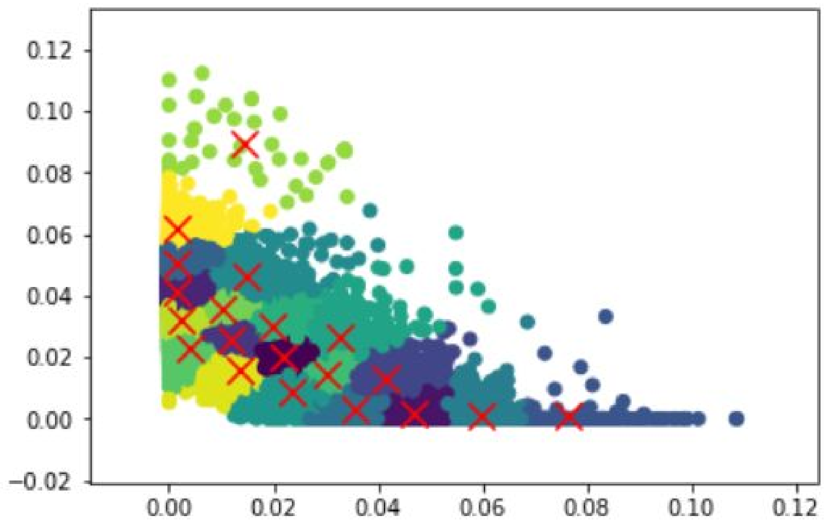

Homogeneity: 0.194
Completeness: 0.206
V-measure: 0.199
Adjusted Rand-Index: 0.061
Adjusted Mutual info score: 0.191

4. <u>Normalized NMF</u>



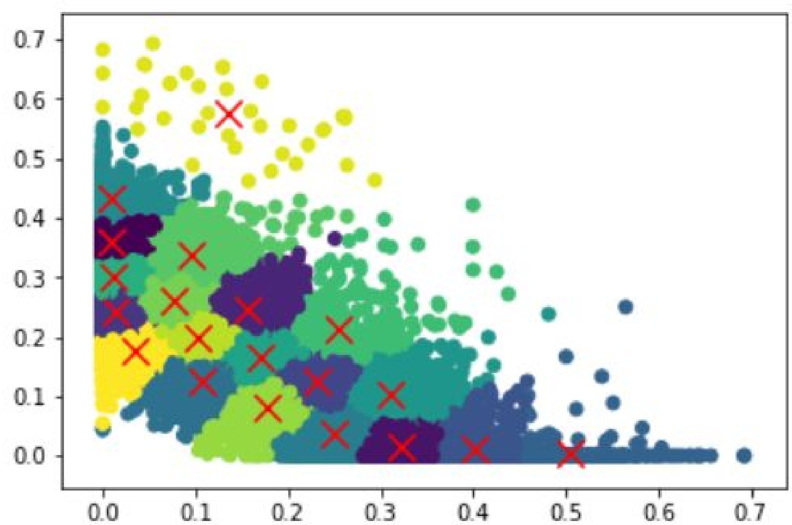

Homogeneity: 0.180
Completeness: 0.188
V-measure: 0.184
Adjusted Rand-Index: 0.058
Adjusted Mutual info score: 0.177

5. <u>Logarithm NMF</u>



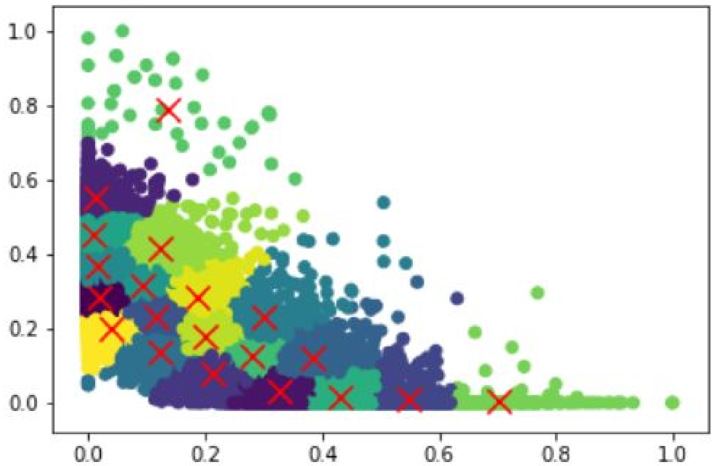

Homogeneity: 0.179
Completeness: 0.188
V-measure: 0.184
Adjusted Rand-Index: 0.058
Adjusted Mutual info score: 0.177

6. <u>Normalization + Logarithm</u>




Homogeneity: 0.181
Completeness: 0.188
V-measure: 0.184
Adjusted Rand-Index: 0.058
Adjusted Mutual info score: 0.178

7. Logarithm + Normalization




Homogeneity: 0.179
Completeness: 0.188
V-measure: 0.183
Adjusted Rand-Index: 0.058
Adjusted Mutual info score: 0.177

**Discussion**:

Dimensional reduction to r=2 improved performance as we expected. With normalization, the results are a little bit better than the base case, but not markedly so. This is likely due to the scaling issues mentioned before. However, the effect is not so pronounced here, possibly due to a larger dimension, as a high amount of scaling in one dimension may not affect the overall clustering result if enough other dimensions are not heavily scaled.

What is interesting to note here is that logarithmic scaling gives us worse results than in part 4. This is possibly the case due to the results of standard statistical tests performed on log-transformed data are often not relevant for the original, non-transformed data.

**Conclusion**:

In this project, we try different methods of clustering, and some transformation to improve the clustering method. However, the results shows that, some transformation such as logarithm transformation is not suit for all the case even though people have the common belief that the log transformation can decrease the variability of data and make data conform more closely to the normal distribution. But it also introduce the problem that the results of standard statistical tests performed on log-transformed data are often not relevant for the original, non-transformed data. So it might make the condition worse. Thus, when we are doing a real application, we need to carefully monitor all the condition and see whether the clustering agree with the reality.