EE219: Project 2

Due on Feb. 12, 2018

Zeyu Zhang (505030513)
Yunchu Zhang (805030502)

Introduction:

Clustering algorithms are unsupervised methods for finding groups of data points that have similar representations in a proper space. K-means clustering is a simple and popular clustering algorithm. In this project, we need to:

1. To find proper representations of the data, s.t. the clustering is efficient and gives out reasonable results.
2. To perform K-means clustering on the dataset, and evaluate the performance of the clustering.
3. To try different preprocess methods which may increase the performance of the clustering.

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. We then use class labels as the ground truth to evaluate the performance of the clustering task.

To get started with a simple clustering task, we take all the documents in the following classes: class 1(com) class 2(rec).

1. Building the TF-IDF matrix

We transform the documents into TF-IDF vectors using min_df=3 and exclude the stopwords. The dimension of the TF-IDF matrix is (7882, 18445)

2. 2-class Clustering

In this part, we apply K-means clustering to classify TF-IDF data into 2 classes. And then we examine the result with homogeneity score, completeness score, V-measure, adjusted Rand score and adjusted mutual info score.

Table 1 K-means clustering with k = 2

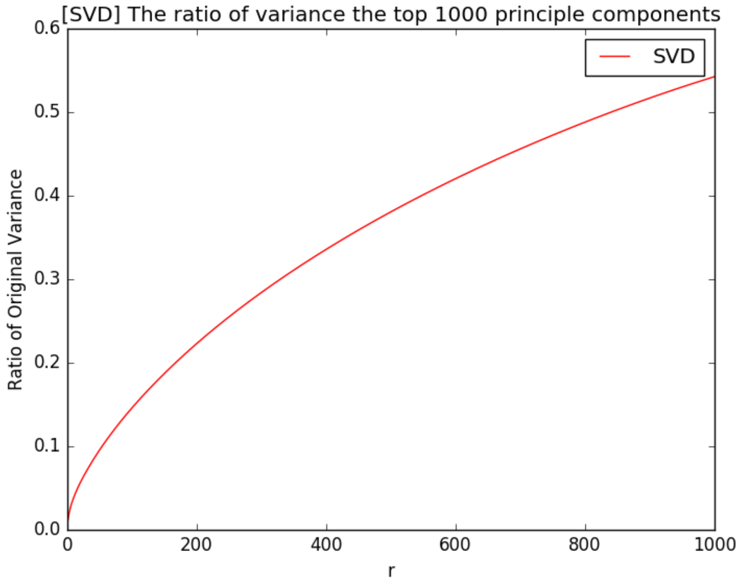
Homogeneity	0.4174
Completeness	0.4582
V-measure	0.4369
Adjusted Rand	0.4186
Adjusted Mutual	0.4173

3. Preprocess the data

For the high dimensional sparse TF-IDF vectors, they cannot yield a good result. Also, when the clusters are not round-shaped, K-means may fail to identify the clusters properly. Thus, we use the package in sklearn – Demonstration of k-means assumptions.

To reduce the dimension, we use NMF and LSI method to dimensionality reduction. Through SVD we calculate the variance remained after dimensionality reduction and sweep over parameters for each method, and choose one that yields better results in clustering purity metrics.

Firstly, we plot the ratio of variance of the original data retained after dimensionality reduction.



We can see that with the increase of the dimension, matrix will contain more information of the original TF-IDF matrix.

Then, we try r=1,2,3,5,10,20,50,100,300 to find the best one for LSI and NMF result

LSI:

R	1	2	3	5	10	20	50	100	300
Homogeneity	0.0792	0.4105	0.408	0.3892	0.4027	0.4099	0.4032	0.4051	0.3974
Completeness	0.0816	0.4474	0.4454	0.4395	0.4452	0.4529	0.4477	0.4492	0.443
V-measure	0.0804	0.4282	0.4259	0.4128	0.4229	0.4303	0.4242	0.426	0.419
Adjusted Rand	0.1042	0.4219	0.4186	0.3705	0.401	0.4065	0.3969	0.3994	0.3892
Adjusted Mutual	0.0791	0.4104	0.408	0.3891	0.4026	0.4099	0.4031	0.4051	0.3974

Contingency matrix is as following

	[LSI] r = 1			[LSI] r = 2			[LSI] r = 3	
	cluster_0	cluster_1		cluster_0	cluster_1		cluster_0	cluster_1
class_0	2190	1713	class_0	2571	1332	class_0	1342	2561
class_1	955	3024	class_1	49	3930	class_1	3930	49
	[LSI] r = 5			[LSI] r = 10			[LSI] r = 20	
	cluster_0	cluster_1		cluster_0	cluster_1		cluster_0	cluster_1
class_0	2376	1527	class_0	2490	1413	class_0	2501	1402
class_1	15	3964	class_1	32	3947	class_1	26	3953
	[LSI] r = 50			[LSI] r = 100			[LSI] r = 300	
	cluster_0	cluster_1		cluster_0	cluster_1		cluster_0	cluster_1
class_0	2470	1433	class_0	1425	2478	class_0	1457	2446
class_1	25	3954	class_1	3954	25	class_1	3954	25

NMF:

R	1	2	3	5	10	20	50	100	300
Homogeneity	0.0792	0.4418	0.0405	0.2721	0.4145	0.3822	0.3499	0.0688	0
Completeness	0.0816	0.44524	0.1382	0.3213	0.4375	0.4238	0.4074	0.1777	0
V-measure	0.0804	0.447	0.0627	0.2947	0.4257	0.4019	0.3765	0.0992	0
Adjusted Rand	0.1042	0.5116	0.0107	0.26	0.46	0.3842	0.3204	0.0185	0
Adjusted Mutual	0.0791	0.4418	0.0404	0.2721	0.4144	0.3821	0.3499	0.0687	0.0001

Contingency matrix is as following

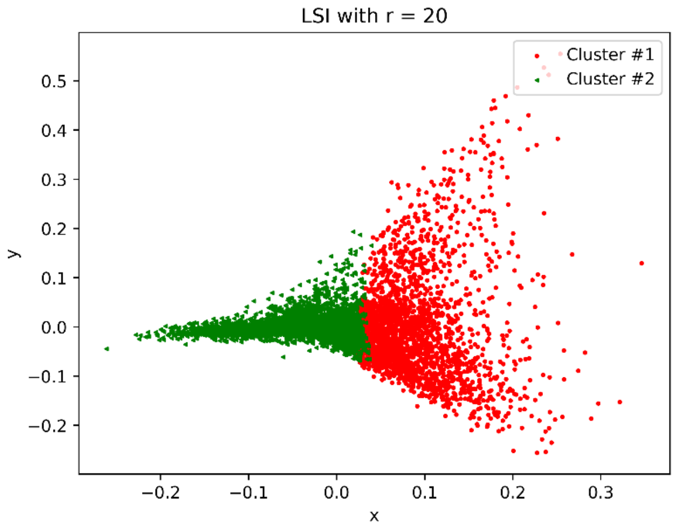
	[NMF] r = 1			[NMF] r = 2			[NMF] r = 3	
	cluster_0	cluster_1		cluster_0	cluster_1		cluster_0	cluster_1
class_0	2190	1713	class_0	3715	188	class_0	3514	389
class_1	955	3024	class_1	934	3045	class_1	3961	18
	[NMF] r = 5			[NMF] r = 10			[NMF] r = 20	
	cluster_0	cluster_1		cluster_0	cluster_1		cluster_0	cluster_1
class_0	2065	1838	class_0	2758	1145	class_0	1449	2454
class_1	93	3886	class_1	123	3856	class_1	3930	49
	[NMF] r = 50			[NMF] r = 100			[NMF] r = 300	
	cluster_0	cluster_1		cluster_0	cluster_1		cluster_0	cluster_1
class_0	1693	2210	class_0	11	3892	class_0	3796	107
class_1	3962	17	class_1	586	3393	class_1	3873	106

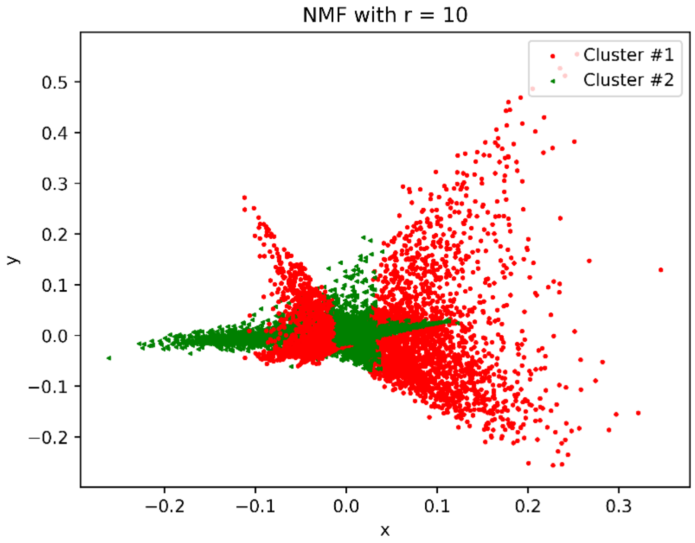
From the results we could see that r=20 and r=10 are the best result for LSI and NMF. With the increase of r (more dimension), the important values are included in the new matrix, but when it goes through the threshold, the more dimension will have little effect on result. That is because the dimension we added are not so important in TF-IDF matrix, thus the result shows non-monotonic behavior of the measures as r increases.

4. Normalization & Non-linear Transform

First, we visualize the performance of the case with the best clustering result. And then, based on the best r we got, we used 3 methods to see whether they increase the clustering performance. Firstly, we use normalization and then non-linear transformation and the combination of both.

We plot the best clustering result in previous part by projecting final data vectors onto 2-dimensional plane and color- coding the classes.





Through the plot, we can see that 2 clusters are overlapping. Thus, we need to try some method to scatter them.

1) LSI with normalization

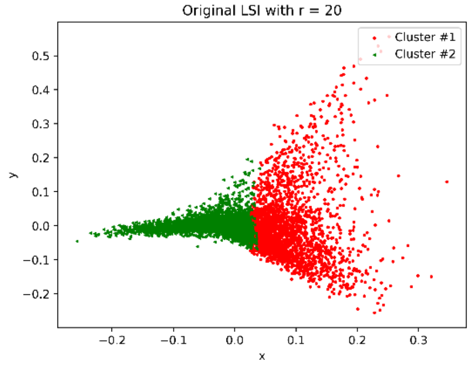
Table 2 LSI with r = 20

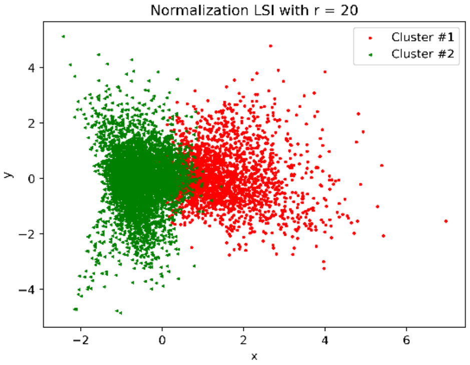
	Origin	Normalization
Homogeneity	0.4025	0.3483
Completeness	0.4480	0.4069
V-measure	0.4240	0.3754
Adjusted Rand	0.3937	0.3167
Adjusted Mutual	0.4024	0.3483

Contingency matrix:

Table 3 Contingency matrix

	Origin		Normalization	
	cluster_0	cluster_1	cluster_0	cluster_1
class_0	2457	1446	2195	1708
class_1	22	3957	16	3964





Although, normalizing on LSI nearly has no impact on purity, it scatters 2 clusters a lot.

NMF with logarithm and normalizing:

Table 4 NMF with logarithm and normalizing

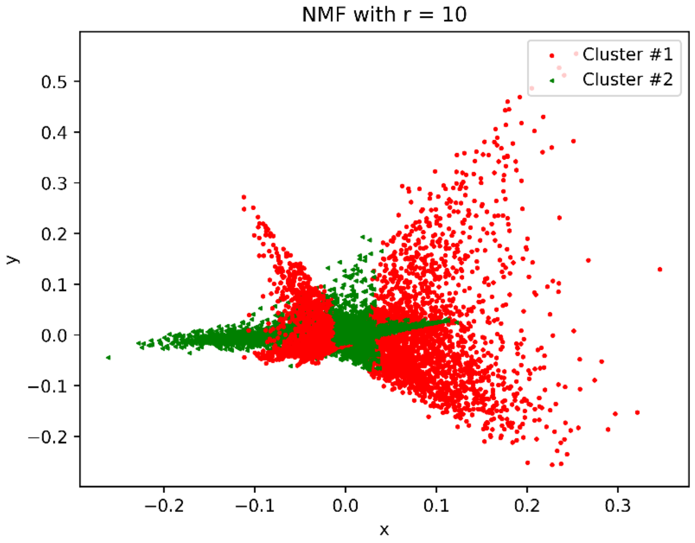
	Origin	Normalization	Logarithm	L+N	N+L
Homogeneity	0.4145	0.4740	0.1871	0.2043	0.0818
Completeness	0.4375	0.4943	0.1899	0.2054	0.0833
V-measure	0.4257	0.4840	0.1885	0.2048	0.0825
Adjusted Rand	0.4600	0.5259	0.2397	0.2658	0.1072
Adjusted Mutual	0.4144	0.4740	0.1870	0.2042	0.0817

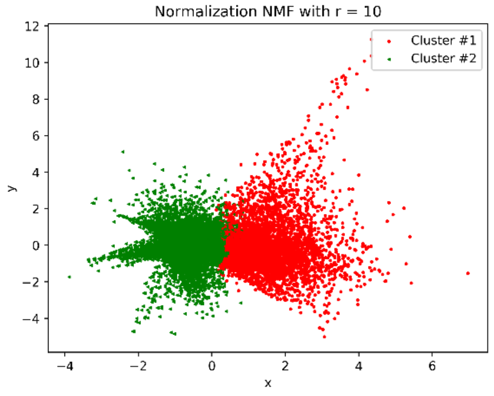
Contingency matrix:

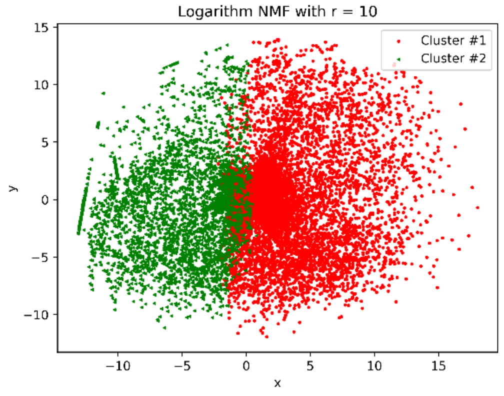
Table 5 Contingency matrix

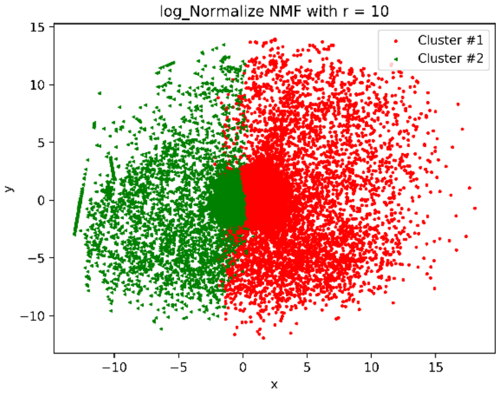
	Origin		Normalization	
	cluster_0	cluster_1	cluster_0	cluster_1
class_0	2785	1145	2912	991
class_1	123	3856	92	3887

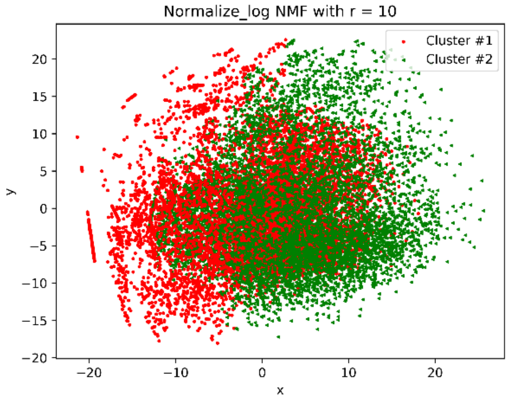
	Logarithm		L+N		N+L	
	cluster_0	cluster_1	cluster_0	cluster_1	cluster_0	cluster_1
class_0	3203	700	3140	763	1001	2902
class_1	1331	2668	1146	2833	2330	1649











From the result above, with normalization on NMF, the purity increased a lot. With logarithm, the 2 clusters could scatter a lot.

With the logarithm transformation, it could reduce the range of data and make the variance of 2 clusters more balanced, which is better for k-means algorithm.

5. Multi- class Clustering

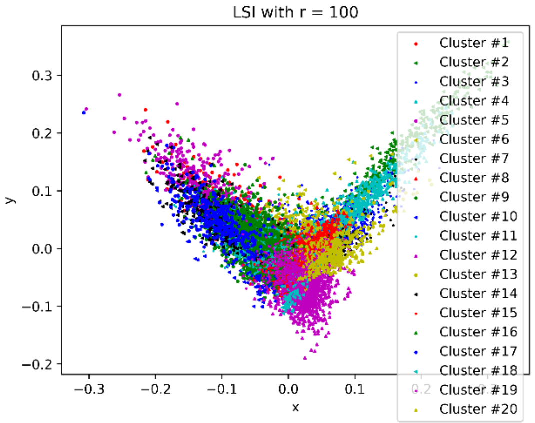
In this part, we include all the documents and the corresponding terms in the data matrix and find proper representation through dimensionality reduction of the TF-IDF representation. We try different dimensionality reduction techniques and transformations.

Firstly, we exam the best r when k-cluster is 20. We found that r=20,10 is not the best parameter when we need to do the high-level clustering. We found the best for LSI and NMF is the same parameter—r =100. (we record the result in task 5 a.txt which will be uploaded)

Secondly, we plot the origin clustering result and new clustering with different transformations and output the contingency matrix.

R=100

LSI with normalization



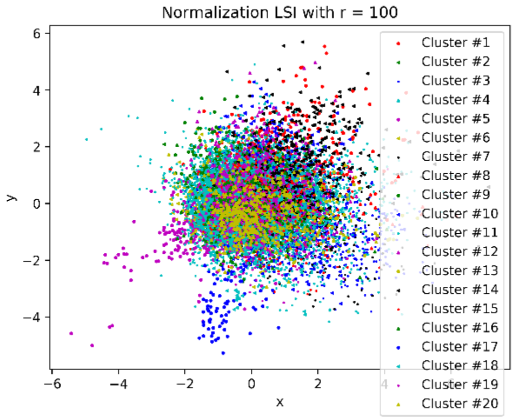


Table 6 LSI with normalization

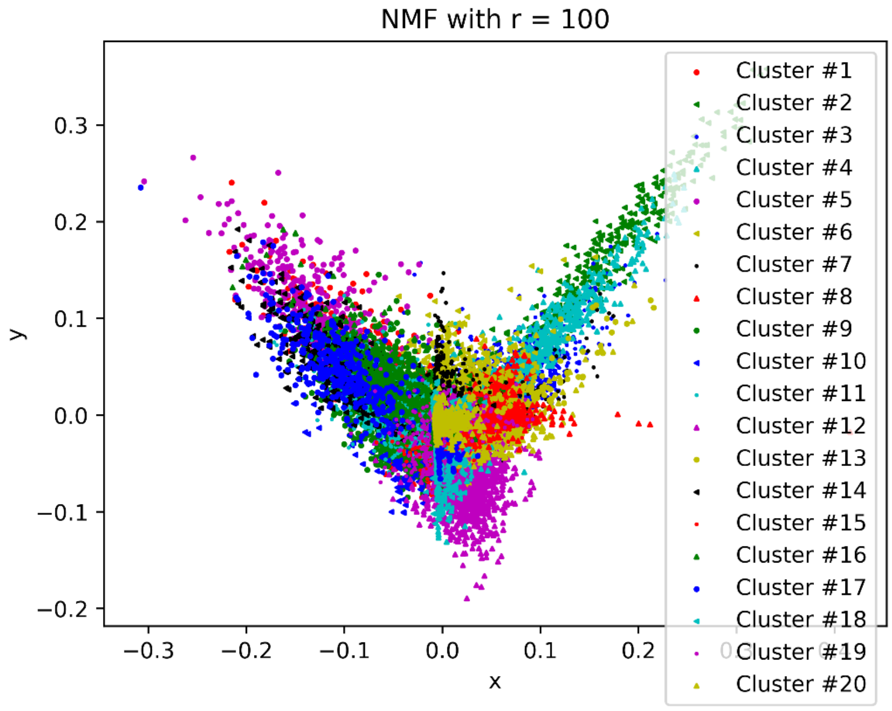
	Origin	Normalization
Homogeneity	0.2871	0.243
Completeness	0.3915	0.3979
V-measure	0.3313	0.3018
Adjusted Rand	0.0595	0.0276
Adjusted Mutual	0.2848	0.2406

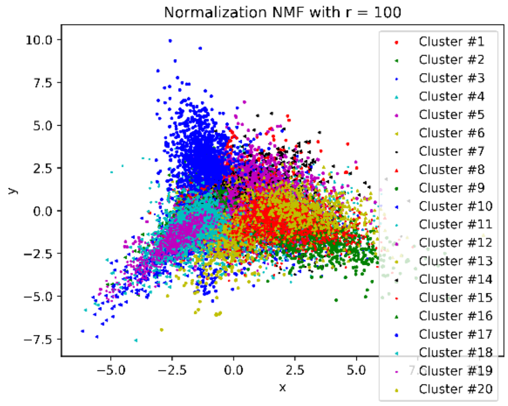
Contingency matrix:

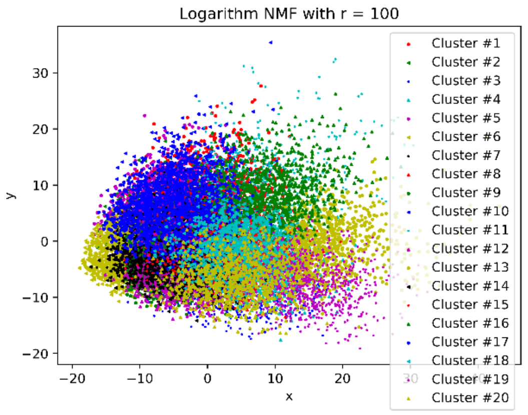
Since the contingency matrix is too large, we do NOT place it here. Please see the appendix.

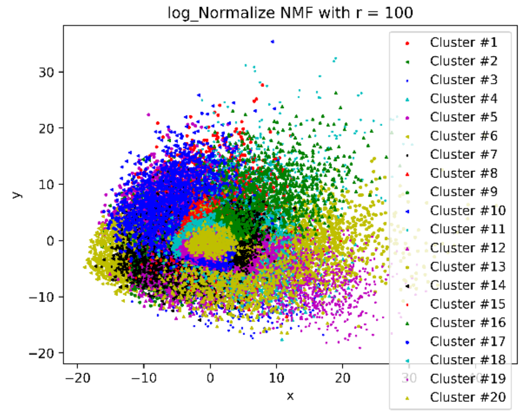
Although, normalizing on LSI nearly has no impact on purity, it scatters 20 clusters a lot.

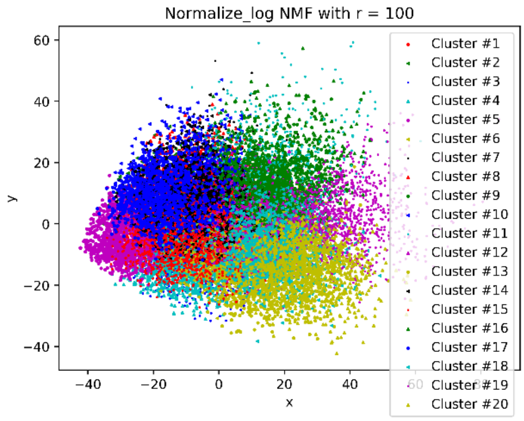
NMF with logarithm and normalizing


NMF with r = 100
0.3
0.2
0.1
0.0
−0.1
−0.2
y
−0.3
−0.2
−0.1
0.0
0.1
0.2
x
Cluster #1
Cluster #2
Cluster #3
Cluster #4
Cluster #5
Cluster #6
Cluster #7
Cluster #8
Cluster #9
Cluster #10
Cluster #11
Cluster #12
Cluster #13
Cluster #14
Cluster #15
Cluster #16
Cluster #17
Cluster #18
Cluster #19
Cluster #20


Normalization NMF with r = 100
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
y
−5.0
−2.5
0.0
2.5
5.0
x
Cluster #1
Cluster #2
Cluster #3
Cluster #4
Cluster #5
Cluster #6
Cluster #7
Cluster #8
Cluster #9
Cluster #10
Cluster #11
Cluster #12
Cluster #13
Cluster #14
Cluster #15
Cluster #16
Cluster #17
Cluster #18
Cluster #19
Cluster #20


Logarithm NMF with r = 100
30
20
10
0
−10
−20
y
−20
−10
0
10
20
x
Cluster #1
Cluster #2
Cluster #3
Cluster #4
Cluster #5
Cluster #6
Cluster #7
Cluster #8
Cluster #9
Cluster #10
Cluster #11
Cluster #12
Cluster #13
Cluster #14
Cluster #15
Cluster #16
Cluster #17
Cluster #18
Cluster #19
Cluster #20


log_Normalize NMF with r = 100
30
20
10
0
−10
−20
y
−20
−10
0
10
20
x
Cluster #1
Cluster #2
Cluster #3
Cluster #4
Cluster #5
Cluster #6
Cluster #7
Cluster #8
Cluster #9
Cluster #10
Cluster #11
Cluster #12
Cluster #13
Cluster #14
Cluster #15
Cluster #16
Cluster #17
Cluster #18
Cluster #19
Cluster #20

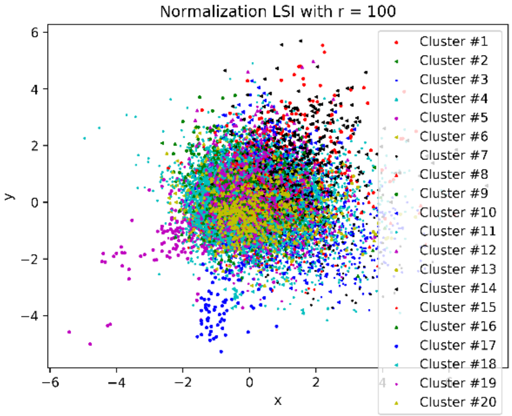

Normalize_log NMF with r = 100
60
40
20
0
−20
−40
y
−40
−20
0
20
40
x
Cluster #1
Cluster #2
Cluster #3
Cluster #4
Cluster #5
Cluster #6
Cluster #7
Cluster #8
Cluster #9
Cluster #10
Cluster #11
Cluster #12
Cluster #13
Cluster #14
Cluster #15
Cluster #16
Cluster #17
Cluster #18
Cluster #19
Cluster #20

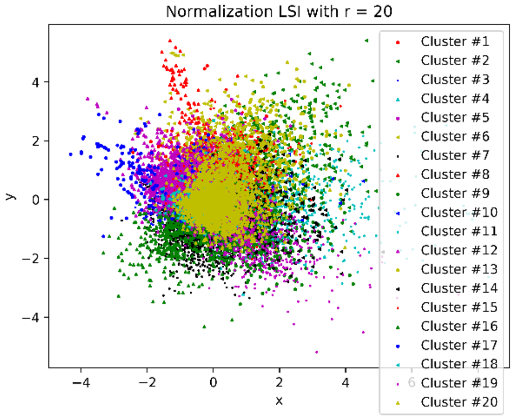
	Origin	Normalization	Logarithm	L+N	N+L
Homogeneity	0.0792	0.2892	0.2146	0.2703	0.1672
Completeness	0.1362	0.4236	0.2169	0.2751	0.169
V-measure	0.1001	0.3437	0.2157	0.2727	0.1681
Adjusted Rand	0.007	0.0599	0.0978	0.1346	0.0718
Adjusted Mutual	0.0761	0.2868	0.2121	0.2679	0.1645

Contingency matrix:
Since the contingency matrix is too large, we do NOT place it here. Please see the appendix.

From the result above, with normalization and logarithm on NMF, the purity increased a lot. With logarithm, the 20 clusters could scatter a lot.

R=20,10
LSI with normalization





	Norm 20	Norm 100
Homogeneity	0.2534	0.243
Completeness	0.3082	0.3979
V-measure	0.2781	0.3018
Adjusted Rand	0.0629	0.0276
Adjusted Mutual	0.251	0.2406

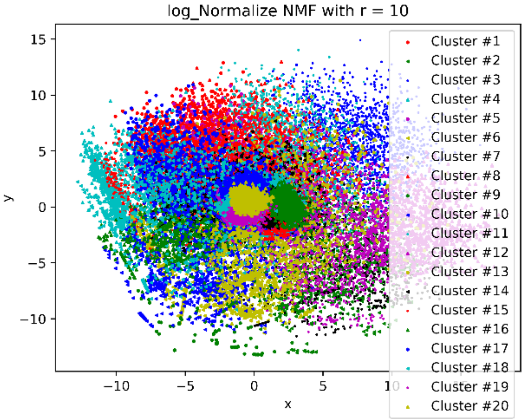
Contingency matrix:
Since the contingency matrix is too large, we do NOT place it here. Please see the appendix.

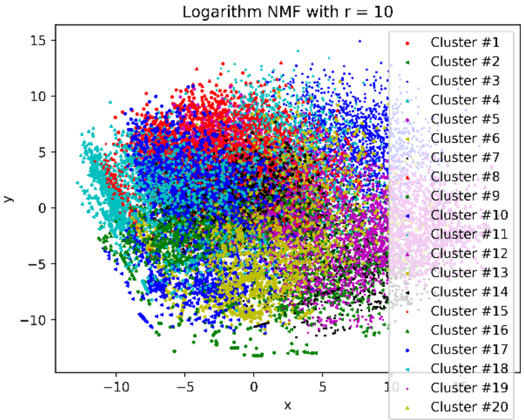
From the result, we could see that when r=100 it’s result is better than r=20 and in the plot parameter 100 scatters 20 clusters a lot.

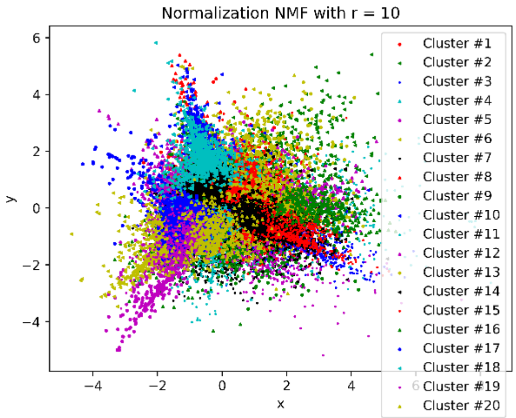
NMF with logarithm and normalizing

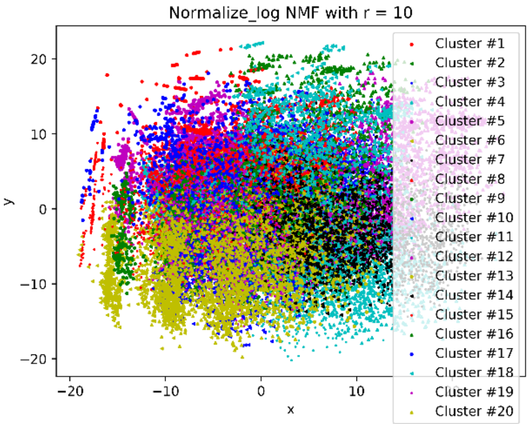
Table 7 r=100

	Origin	Normalization	Logarithm	L+N	N+L
Homogeneity	0.0792	0.2892	0.2146	0.2703	0.1672
Completeness	0.1362	0.4236	0.2169	0.2751	0.169
V-measure	0.1001	0.3437	0.2157	0.2727	0.1681
Adjusted Rand	0.007	0.0599	0.0978	0.1346	0.0718
Adjusted Mutual	0.0761	0.2868	0.2121	0.2679	0.1645









	Origin	Normalization	Logarithm	L+N	N+L
Homogeneity	0.0920	0.2643	0.1804	0.1966	0.1362
Completeness	0.1499	0.3097	0.1817	0.1977	0.1369
V-measure	0.1141	0.2852	0.1810	0.1971	0.1365
Adjusted Rand	0.0068	0.0815	0.0703	0.0791	0.0458
Adjusted Mutual	0.0890	0.2619	0.1778	0.1940	0.1334

Contingency matrix:
Since the contingency matrix is too large, we do NOT place it here. Please see the appendix.

From the result above, we can see the parameter 100 is better than 10 for all the transformations and the purity increased a lot within each parameter when applying transformations. That may because when the dimension of cluster increases, it need more row/column's information to make a better clustering. Thus, r=100 is better than r=10,20.

Appendix

Original LSI with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	5	84	283	51	0	2	2	353	0	0	9	1	6	0	0	1	0	0	2	0
class_1	206	0	31	0	29	2	0	341	138	4	150	0	0	0	0	22	50	0	0	0
class_2	143	1	24	0	296	1	0	263	79	1	62	0	0	16	0	26	73	0	0	0
class_3	17	0	7	0	42	3	0	187	289	1	93	3	0	179	0	4	155	0	2	0
class_4	10	0	6	0	4	1	0	226	469	1	90	1	0	98	0	2	55	0	0	0

class_5	56	0	13	0	50	9	0	324	33	5	104	0	0	1	0	384	9	0	0	0
class_6	0	0	13	0	12	0	0	263	433	62	54	16	0	52	3	1	39	0	27	0
class_7	1	0	55	0	0	1	0	539	22	2	35	0	1	3	0	4	0	0	327	0
class_8	0	2	56	0	0	0	0	610	7	0	26	2	0	12	256	1	0	0	24	0
class_9	2	1	40	0	0	1	0	529	0	2	43	375	0	0	0	0	0	0	1	0
class_10	0	0	30	0	0	0	0	335	3	8	22	600	0	0	0	0	0	0	1	0
class_11	15	0	184	0	5	339	0	366	50	1	20	0	9	0	0	1	0	0	1	0
class_12	11	1	21	0	4	6	0	576	235	1	85	2	0	12	1	2	10	0	17	0
class_13	2	4	193	0	1	0	0	649	6	0	63	0	1	0	0	0	0	71	0	0
class_14	16	0	161	0	0	0	0	749	25	1	32	2	0	0	0	0	0	0	1	0
class_15	1	179	117	404	1	0	1	268	1	0	22	0	3	0	0	0	0	0	0	0
class_16	3	3	345	4	0	1	2	352	0	3	6	1	185	0	0	1	1	0	3	0
class_17	0	5	166	3	0	0	271	324	0	2	4	0	3	0	0	0	0	0	0	162
class_18	0	1	367	0	0	0	1	336	0	0	8	1	58	1	0	0	0	0	2	0
class_19	1	48	116	119	0	0	2	283	0	0	6	0	50	0	0	0	0	1	2	0

Original NMF with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	5	8	45	99	0	1	0	45	0	0	1	447	19	0	82	0	0	0	8	39
class_1	75	28	37	2	0	71	20	90	5	0	0	438	24	124	5	0	0	12	20	22
class_2	48	96	48	2	0	114	13	59	4	0	0	471	37	15	6	5	1	11	19	36
class_3	87	79	29	0	0	28	3	72	14	0	1	452	20	12	8	0	0	115	27	35
class_4	30	81	33	0	0	11	3	90	18	0	3	456	17	27	12	0	0	115	32	35
class_5	58	39	51	1	0	21	323	50	1	0	0	347	26	31	6	0	0	2	9	23
class_6	10	5	18	0	4	4	1	39	25	0	5	733	6	6	3	0	0	20	3	93
class_7	14	43	49	1	0	2	2	84	3	0	0	663	17	1	15	0	0	7	29	60
class_8	15	21	70	0	0	3	1	76	0	0	4	684	27	1	4	0	0	15	17	58
class_9	13	6	45	2	67	2	0	64	0	0	0	686	9	0	35	0	0	0	23	42
class_10	12	5	56	2	157	0	0	52	0	0	3	613	12	0	18	0	0	2	23	44
class_11	8	13	78	1	0	18	5	73	1	0	1	637	17	9	36	0	0	0	11	83
class_12	56	23	51	0	0	15	0	95	9	0	2	589	37	9	8	0	0	23	13	54
class_13	33	24	45	19	0	3	0	83	0	71	1	548	33	5	32	0	43	0	18	32
class_14	12	19	53	4	0	18	1	60	0	0	3	671	17	6	20	23	0	0	28	52
class_15	8	10	79	33	0	3	0	72	0	0	0	522	13	3	197	0	0	0	17	40
class_16	10	9	62	1	0	1	0	56	0	0	0	604	15	0	55	1	1	0	31	64
class_17	9	8	57	9	1	0	0	53	0	0	0	705	18	0	31	0	0	0	13	36
class_18	3	12	42	10	2	1	0	46	0	0	1	515	17	0	45	0	0	1	16	64
class_19	3	3	46	56	0	3	1	41	1	2	0	357	9	1	64	0	0	2	15	24

Normalization LSI with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	7	2	89	2	33	0	0	0	1	0	637	5	16	2	0	1	0	0	0	4
class_1	0	0	0	0	17	0	23	0	3	34	559	57	0	2	197	47	0	5	16	13
class_2	0	0	1	0	27	0	137	0	0	65	417	48	0	3	72	171	0	1	32	11
class_3	0	2	0	0	31	0	58	0	2	372	382	59	0	5	15	53	0	1	1	1
class_4	0	0	0	0	12	0	1	0	1	481	371	50	0	2	5	34	0	1	1	4

class_5	0	0	0	0	19	0	13	0	1	7	448	35	0	11	30	10	0	5	404	5
class_6	1	33	0	0	28	0	23	0	19	121	563	102	0	1	1	21	0	59	1	2
class_7	6	321	0	0	16	0	0	0	0	2	606	25	0	1	0	6	0	2	5	0
class_8	1	283	4	0	29	1	0	0	3	9	639	16	2	0	0	6	0	0	2	1
class_9	0	1	2	0	25	224	0	0	103	0	605	25	0	1	5	0	0	2	1	0
class_10	2	1	0	0	1	421	0	0	202	0	340	22	0	0	0	0	0	8	0	2
class_11	15	0	0	0	40	1	8	0	0	7	530	25	0	354	2	4	0	1	0	4
class_12	2	14	1	0	23	0	2	0	3	35	824	46	0	6	4	15	0	1	1	7
class_13	1	1	4	0	11	0	1	0	0	1	876	19	2	0	2	1	71	0	0	0
class_14	1	0	0	0	9	3	0	0	1	1	587	16	0	0	14	1	0	1	0	353
class_15	7	0	217	0	9	0	0	0	0	0	686	16	58	0	1	0	0	0	0	3
class_16	360	2	4	2	50	1	0	0	1	0	473	8	1	0	0	0	0	3	1	4
class_17	3	0	7	272	5	1	0	162	0	0	481	6	1	0	0	0	0	2	0	0
class_18	66	1	3	1	15	0	0	0	1	1	589	6	90	0	0	0	0	1	0	1
class_19	66	2	66	2	6	1	0	0	2	0	458	6	13	0	0	3	2	0	0	1

Normalization NMF with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	386	6	3	23	0	0	2	0	0	265	0	4	2	1	12	0	0	38	57	0
class_1	345	0	0	0	6	0	0	0	246	0	9	10	5	0	344	0	0	0	0	8
class_2	286	0	0	0	6	0	0	5	459	3	12	6	1	0	170	0	0	3	0	34
class_3	247	0	0	0	29	0	0	0	72	1	59	0	7	0	302	0	3	0	0	262
class_4	342	0	1	0	46	0	0	0	20	1	80	3	6	2	344	0	1	1	0	116
class_5	373	0	0	0	8	0	0	0	403	2	4	3	11	0	182	0	1	0	0	1
class_6	256	0	0	0	508	2	23	0	8	0	23	1	1	2	80	0	14	1	0	56
class_7	569	0	0	0	32	1	313	0	3	3	0	0	2	0	43	0	1	20	0	3
class_8	650	0	0	0	12	243	23	0	0	2	0	1	0	1	32	0	2	18	2	10
class_9	458	6	3	0	4	0	1	0	1	4	0	0	1	0	37	0	479	0	0	0
class_10	378	0	0	0	8	0	1	0	0	2	0	1	0	1	15	0	587	6	0	0
class_11	413	0	1	0	4	0	1	0	24	2	8	3	360	0	44	0	2	128	0	1
class_12	685	0	0	0	21	0	17	0	25	0	9	7	20	1	173	0	2	8	0	16
class_13	820	0	0	0	0	0	0	0	3	15	0	0	0	0	72	71	1	4	4	0
class_14	563	0	0	0	4	0	1	23	5	2	1	328	0	1	35	0	2	22	0	0
class_15	344	4	0	0	0	0	0	0	2	195	0	2	0	0	19	0	1	19	410	1
class_16	396	2	0	0	3	0	2	1	2	9	0	2	0	0	7	0	1	482	3	0
class_17	390	296	3	0	1	0	0	0	0	35	0	0	0	0	6	0	2	204	3	0
class_18	450	1	0	0	3	0	2	0	0	16	0	1	0	0	9	0	3	238	52	0
class_19	300	8	7	0	0	0	2	0	2	138	0	1	1	0	7	1	0	61	100	0

Logarithm NMF with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	9	2	105	38	4	68	15	21	91	0	6	201	41	2	2	5	9	32	97	51

class_1	132	26	4	19	182	74	89	17	37	39	86	54	23	3	1	54	106	17	10	0
class_2	222	25	0	19	228	99	63	3	32	67	38	39	8	3	2	56	62	13	4	2
class_3	107	49	0	16	68	79	69	7	17	336	16	30	3	1	3	103	61	11	4	2
class_4	50	116	2	19	42	81	88	15	44	254	8	34	10	3	1	114	55	23	2	2
class_5	423	16	0	11	114	57	68	13	31	11	95	35	5	0	0	20	75	4	5	5
class_6	7	408	3	13	33	66	36	48	21	111	16	8	11	11	0	50	113	14	3	3
class_7	10	114	7	133	13	98	84	55	92	3	11	63	49	2	2	84	50	84	11	25
class_8	11	89	3	196	7	86	53	60	146	4	8	52	70	4	1	58	25	87	16	20
class_9	7	20	7	174	2	103	42	53	79	0	0	44	17	367	0	2	34	23	14	6
class_10	6	12	2	191	0	83	34	66	53	0	3	30	15	449	0	4	23	6	15	7
class_11	20	20	3	21	32	78	20	25	57	18	54	48	39	3	342	49	30	25	48	59
class_12	41	85	2	31	23	55	71	33	34	64	46	23	48	5	9	236	85	76	4	13
class_13	9	5	3	18	4	81	45	38	61	1	36	50	188	1	0	17	57	348	20	8
class_14	15	32	12	93	20	79	51	87	117	5	62	55	104	3	10	51	44	97	18	32
class_15	2	4	455	8	4	54	25	35	82	0	13	71	41	2	2	2	34	16	123	24
class_16	6	11	20	62	3	69	37	79	122	1	12	72	53	2	15	9	14	59	81	183
class_17	2	8	14	31	1	82	35	79	141	0	8	111	35	3	1	4	16	27	103	239
class_18	4	10	14	38	2	47	33	68	72	2	10	90	54	5	7	5	11	37	113	153
class_19	2	3	134	25	0	58	26	21	63	0	4	108	30	4	2	1	16	27	76	28

Logarithm + Normalization NMF with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	27	5	2	115	61	28	107	0	167	7	8	154	4	15	0	10	72	6	9	2
class_1	10	116	22	101	3	17	3	13	52	16	59	7	258	61	90	97	7	23	5	13
class_2	7	66	16	115	3	10	1	25	26	6	293	4	143	38	154	42	2	28	4	2
class_3	6	73	72	76	0	7	1	274	13	20	87	3	23	61	189	24	2	44	2	5
class_4	12	47	445	100	3	14	0	84	27	21	45	6	13	49	46	10	2	33	4	2
class_5	8	127	13	108	7	9	1	2	40	7	369	9	124	27	28	82	2	21	1	3
class_6	3	112	57	75	6	20	0	110	17	52	13	10	11	44	53	21	4	348	16	3
class_7	18	57	3	118	29	54	1	0	46	448	6	10	5	109	16	12	20	32	3	3
class_8	22	35	5	130	30	154	3	2	87	317	9	18	4	100	6	5	26	34	8	1
class_9	17	41	1	153	10	155	5	1	94	25	9	15	3	5	1	3	9	49	397	1
class_10	11	20	2	91	12	176	1	0	47	10	3	9	0	6	0	4	16	21	570	0
class_11	21	30	14	90	75	9	4	9	55	7	17	15	22	52	4	73	56	17	5	416
class_12	27	110	73	78	7	25	0	46	42	89	45	1	23	247	18	56	16	64	5	12
class_13	464	62	4	119	42	50	12	0	64	17	5	20	7	44	2	45	13	16	3	1
class_14	59	60	11	130	58	77	11	6	120	73	17	26	19	109	4	79	45	52	17	14
class_15	15	38	2	66	35	15	329	0	39	4	4	372	4	8	0	17	31	8	8	2
class_16	31	11	0	113	161	70	15	0	156	39	10	29	3	26	4	24	167	23	8	20
class_17	19	16	2	104	125	39	29	0	92	5	4	20	3	9	0	9	439	17	4	4
class_18	39	14	1	89	202	30	16	1	126	9	7	29	0	10	0	16	147	16	13	10
class_19	31	14	3	89	51	15	121	0	72	12	2	156	1	5	1	4	40	2	5	4

Normalization + Logarithm NMF with r = 100

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	13	6	61	187	91	74	3	14	4	0	5	35	131	14	19	12	3	27	33	67
class_1	18	33	7	27	91	44	35	117	30	40	105	15	14	61	9	62	215	11	34	5
class_2	14	42	6	21	108	34	20	247	26	80	43	4	4	62	10	104	126	10	18	6
class_3	21	49	5	12	84	29	40	78	45	294	21	5	2	53	9	125	82	10	16	2
class_4	26	126	8	19	102	39	30	31	68	221	11	12	7	53	10	105	54	22	16	3
class_5	20	20	11	18	76	29	20	318	9	18	101	14	9	67	5	87	132	10	17	7
class_6	32	324	10	8	78	13	10	15	64	126	30	40	5	114	25	14	29	16	16	6
class_7	169	115	36	32	114	54	12	4	123	13	11	40	4	50	11	23	22	61	63	33
class_8	181	83	35	37	104	64	1	10	73	5	9	35	13	48	21	15	21	119	81	41
class_9	40	24	9	27	127	66	15	13	14	2	4	39	15	36	348	3	26	150	15	21
class_10	18	14	18	12	91	54	2	8	13	0	3	50	8	30	422	4	9	207	19	17
class_11	16	24	155	27	106	175	55	22	33	9	55	41	17	44	8	34	28	10	47	85
class_12	83	69	15	9	68	43	55	35	184	38	49	33	6	55	16	71	50	24	74	7
class_13	115	17	28	55	99	75	15	5	37	2	39	88	41	60	9	19	29	42	162	53
class_14	95	35	42	32	98	87	49	16	72	2	41	82	17	59	26	15	24	45	109	41
class_15	9	6	27	302	56	28	1	8	3	3	19	37	309	39	12	3	19	19	49	48
class_16	55	12	170	53	92	83	2	9	20	3	13	92	31	26	15	9	5	46	50	124
class_17	25	13	230	61	98	57	2	5	12	4	9	105	52	29	14	7	14	36	23	144
class_18	20	11	122	55	71	50	17	6	8	3	4	79	28	19	14	3	7	28	38	192
class_19	19	3	38	146	75	33	1	8	3	2	5	33	117	19	13	1	15	15	34	48

Normalization LSI with r = 20

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	1	99	277	11	6	2	1	0	290	0	2	46	0	0	0	2	0	4	12	46
class_1	28	0	210	164	123	208	1	0	45	0	3	66	25	0	56	0	4	0	0	40
class_2	38	1	211	56	56	132	0	0	24	0	1	62	275	16	72	0	1	0	1	39
class_3	14	0	219	48	75	19	3	0	13	0	5	72	39	182	187	2	1	0	0	103
class_4	15	0	318	79	83	8	1	0	24	0	1	72	4	102	117	0	1	0	0	138
class_5	355	0	240	62	89	65	0	0	10	0	8	57	45	1	9	0	4	0	0	43
class_6	0	0	321	28	29	2	25	0	8	0	1	277	17	64	49	38	72	0	0	44
class_7	4	0	384	9	30	1	0	0	41	0	1	35	0	3	0	334	3	0	9	136
class_8	2	3	382	6	25	1	2	0	38	0	0	52	1	13	1	58	0	0	5	407
class_9	1	1	421	11	27	2	330	0	56	0	1	49	0	0	0	1	7	0	0	87
class_10	0	0	315	1	13	1	538	0	30	0	0	32	0	0	1	1	22	0	1	44
class_11	1	0	259	68	13	15	1	0	74	0	327	65	5	0	3	1	2	0	84	73
class_12	5	1	379	157	82	12	2	0	21	0	6	62	3	13	13	18	1	0	5	204
class_13	0	4	422	93	48	1	0	71	122	0	0	37	0	0	0	0	1	0	2	189
class_14	1	0	307	477	21	10	0	0	43	0	0	33	0	0	0	1	1	0	6	87
class_15	0	307	231	17	21	1	0	0	296	0	0	20	1	0	0	0	0	4	7	92
class_16	1	3	257	10	5	3	1	0	57	0	1	38	0	0	1	3	4	2	458	66

class_17	0	5	295	7	4	1	0	0	98	165	0	19	0	0	1	0	3	275	20	47
class_18	0	1	283	50	5	0	1	0	197	0	0	18	0	1	0	2	0	1	158	58
class_19	0	90	236	2	3	1	2	2	169	0	0	14	0	0	0	2	0	5	57	45

Logarithm + Normalization NMF with r = 10

c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19	
class_0	4	146	3	0	165	31	9	3	75	1	20	8	23	3	12	3	130	126	19	18
class_1	45	20	40	174	9	11	13	43	87	5	121	34	38	170	36	106	7	6	3	5
class_2	20	11	57	261	26	15	10	99	63	16	71	10	53	124	31	58	26	15	15	4
class_3	6	8	213	53	5	6	11	195	18	9	60	44	38	68	26	200	5	11	6	0
class_4	8	5	219	24	9	4	20	183	11	3	65	82	46	44	34	170	11	8	13	4
class_5	84	12	31	167	13	17	12	29	141	6	140	7	26	163	17	99	7	5	9	3
class_6	5	10	131	33	23	15	70	135	10	40	39	142	26	33	139	102	6	8	6	2
class_7	40	42	31	7	69	87	300	90	16	10	9	56	60	28	54	17	23	6	8	37
class_8	28	58	26	3	68	152	272	64	19	11	18	40	35	15	45	13	52	11	6	60
class_9	19	90	4	5	158	143	8	15	30	233	23	25	54	4	74	4	51	14	11	29
class_10	15	78	3	0	117	141	5	7	16	385	8	25	27	3	68	2	35	21	11	32
class_11	149	17	0	13	4	13	5	36	125	2	87	17	34	17	13	68	36	36	97	222
class_12	87	20	53	11	15	25	57	108	50	9	79	85	31	69	33	183	12	5	7	45
class_13	17	47	6	4	28	68	22	16	29	3	109	17	36	4	161	6	360	8	11	38
class_14	51	69	11	4	83	85	83	53	78	7	71	49	39	38	46	46	59	6	7	102
class_15	1	208	7	1	173	5	3	8	98	1	32	7	24	3	5	3	154	258	4	2
class_16	1	52	2	5	52	128	3	31	64	1	52	44	27	11	51	3	46	39	131	167
class_17	2	74	5	1	55	151	5	7	27	1	22	29	26	3	38	4	45	76	314	55
class_18	1	65	3	1	74	82	6	15	60	2	46	28	28	6	24	7	58	44	93	132
class_19	0	112	1	2	88	20	9	9	59	5	19	3	27	2	6	5	98	120	21	22

Normalization + Logarithm NMF with r = 10

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19
class_0	47	91	7	43	17	63	23	4	61	103	50	1	27	33	67	6	7	3	32	114
class_1	41	15	91	149	5	3	134	25	0	24	61	137	18	20	4	171	6	39	8	22
class_2	74	24	66	83	14	2	75	12	2	88	44	61	9	71	13	214	24	75	11	23
class_3	51	20	28	38	10	2	150	57	1	16	24	168	52	89	10	100	17	117	23	9
class_4	47	14	11	41	12	4	186	92	6	7	11	139	70	99	15	61	10	104	21	13
class_5	36	25	150	152	6	0	116	5	0	44	113	101	6	39	10	108	14	42	2	19
class_6	28	7	4	29	58	2	40	191	1	2	9	143	127	14	3	62	60	132	49	14
class_7	59	67	34	18	218	51	24	138	71	19	14	20	40	26	8	8	5	91	36	43
class_8	35	52	33	27	192	86	11	111	99	35	20	18	41	29	8	4	3	48	49	95
class_9	50	4	24	34	83	151	6	33	1	87	45	6	49	12	12	2	122	11	79	183
class_10	27	4	21	11	88	152	5	22	3	57	23	5	42	7	14	1	250	5	69	193
class_11	70	40	61	56	28	71	59	4	63	45	115	27	49	40	113	7	18	48	22	55
class_12	31	31	57	66	34	14	117	70	25	18	40	172	73	45	6	13	14	113	25	20

class_13	60	148	16	84	31	57	40	16	92	80	61	7	76	35	17	3	2	11	39	115
class_14	37	63	54	78	59	57	68	37	77	67	66	28	62	56	7	6	3	29	43	90
class_15	79	131	6	59	21	62	41	4	32	81	52	6	73	29	88	21	19	10	67	116
class_16	29	59	1	65	2	85	23	4	67	64	56	0	84	73	132	7	1	0	70	88
class_17	25	84	1	35	5	70	22	3	42	40	21	2	64	26	308	11	2	0	82	97
class_18	26	54	0	48	5	64	29	3	56	52	44	1	74	57	100	6	2	0	75	79
class_19	61	61	1	38	14	59	24	3	34	48	35	3	29	24	58	7	12	1	37	79