EE219: Large-Scale Data Mining: Models and Algorithms
Winter 2018

Project 2: Clustering

Zhi Ming CHUA (UID: 805-068-401)
Vijay RAVI (UID: 805-033-666)

February 9, 2018

Contents

1 Introduction

Clustering algorithms are unsupervised methods for finding groups of data points that have similar representations in a proper space. Clustering differs from classification in that no a priori labeling (grouping) of the data points is available.

2 Dataset and Problem Statement

In this project, the "20 Newsgroups" dataset previously explored in Project 1 is used. It is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups (classes), each corresponding to a different topic.

For the clustering task, the class labels are not available to the clustering algorithm and the aim is to find groupings of the documents, where documents in each group are more similar to each other than to those in other groups. These groups, or clusters, capture the dependencies among the documents that are known through class labels.

The class labels are used as the ground truth to evaluate the performance of the clustering task with the following 5 measure scores:

1. **Homogeneity**: Measures the "purity" of clusters
2. **Completeness**: Measures if all data points of a class are assigned to the same cluster
3. **V-Measure**: Harmonic mean of homogeneity score and completeness. The harmonic mean ensures that the score is low if at least one of homogeneity and completeness performs badly
4. **Rand Index**: Similar to accuracy, measures similarity between clustering labels to ground truth labels
5. **Adjusted Mutual Information**: Measures the mutual information between the cluster label distribution and the ground truth label distribution

3 TFIDF

The dataset is first converted to a TFIDF representation. This is carried out using `sklearn.feature_extraction.text.CountVectorizer` with parameter `min_df=3`.

The size of the TFIDF matrix is (7882, 27768).

4 Dataset Without Dimensionality Reduction

k-means clustering is applied to the dataset with $k = 2$ using `sklearn.cluster.KMeans`. `n_init` is set to 30 to run the algorithm multiple times with different centroid seeds, with the best results in terms of inertia given as the output.

The measure scores are tabulated below.

Homogeneity	0.251
Completeness	0.334
V-Measure	0.286
Adjusted Rand Index	0.177
Adjusted Mutual Information	0.251

Table 1: Measure Scores: Without Dimensionality Reduction

The contingency matrix is shown in Figure 1.

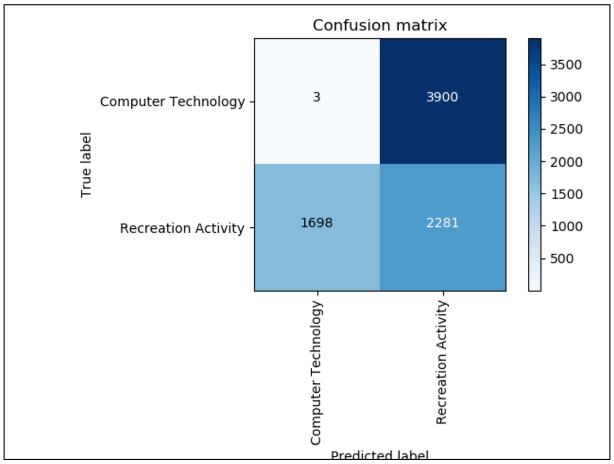


Figure 1: Contingency Matrix

As expected, the original TFIDF matrix does not yield good clustering results.

5 Dataset with Dimensionality Reduction: SVD, NMF

High dimensionality typically does not work well because the volume of the space becomes too large that the data becomes sparse. In this section, Latent Semantic Indexing (LSI) and Non-Negative Matrix Factorization (NMF) are explored as means to reduce the dimensionality of the dataset.

5.1 Variance of Truncated Dataset

The variance of the truncated dataset gives some representation on how much it resembles the original dataset. The proportion of variance retained from the original dataset after the dimensionality reduction via SVD is plotted against varying number of dimensions r. The ratio of variance retained plotted in Figure 2 is for $r = 1$ to 1000.

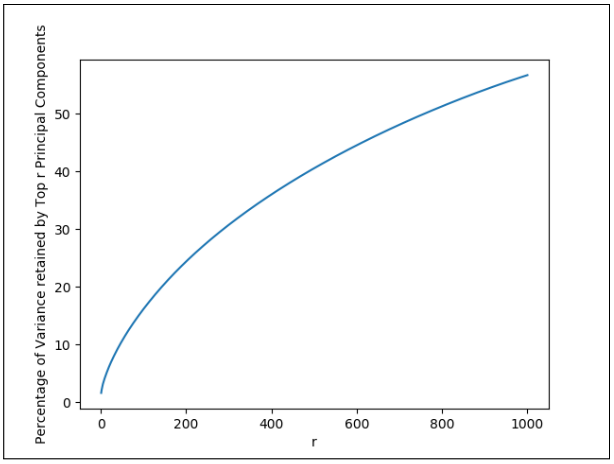


Figure 2: Percentage of Total Variance Against r

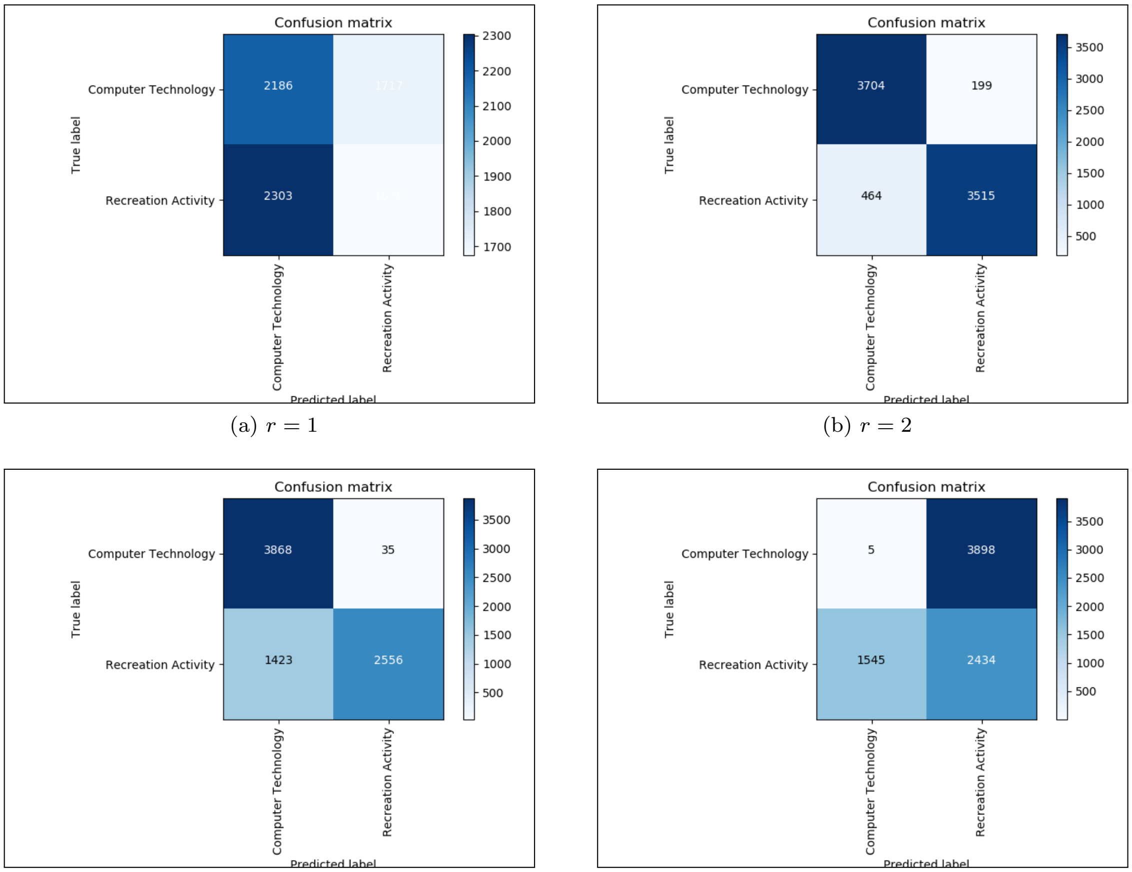
The plot shows that the variance retained strictly increases with the number of dimensions. Also, it can be observed that the curve is concave, which means that the benefit of increasing r is diminishing. For $r = 1000$, 56.7% of the original dataset variance is retained with a 96.4% reduction (27768 to 1000) in dimension of dataset.

5.2 Performance for Varying r

The performance of the clustering will be evaluated for varying r using LSI and NMF. $r = 1, 2, 3, 5, 10, 20, 50, 100, 300$ is investigated in the following sections.

5.2.1 LSI

The contingency matrices are shown in Figure 3 and the measure scores are plotted in Figure 4.



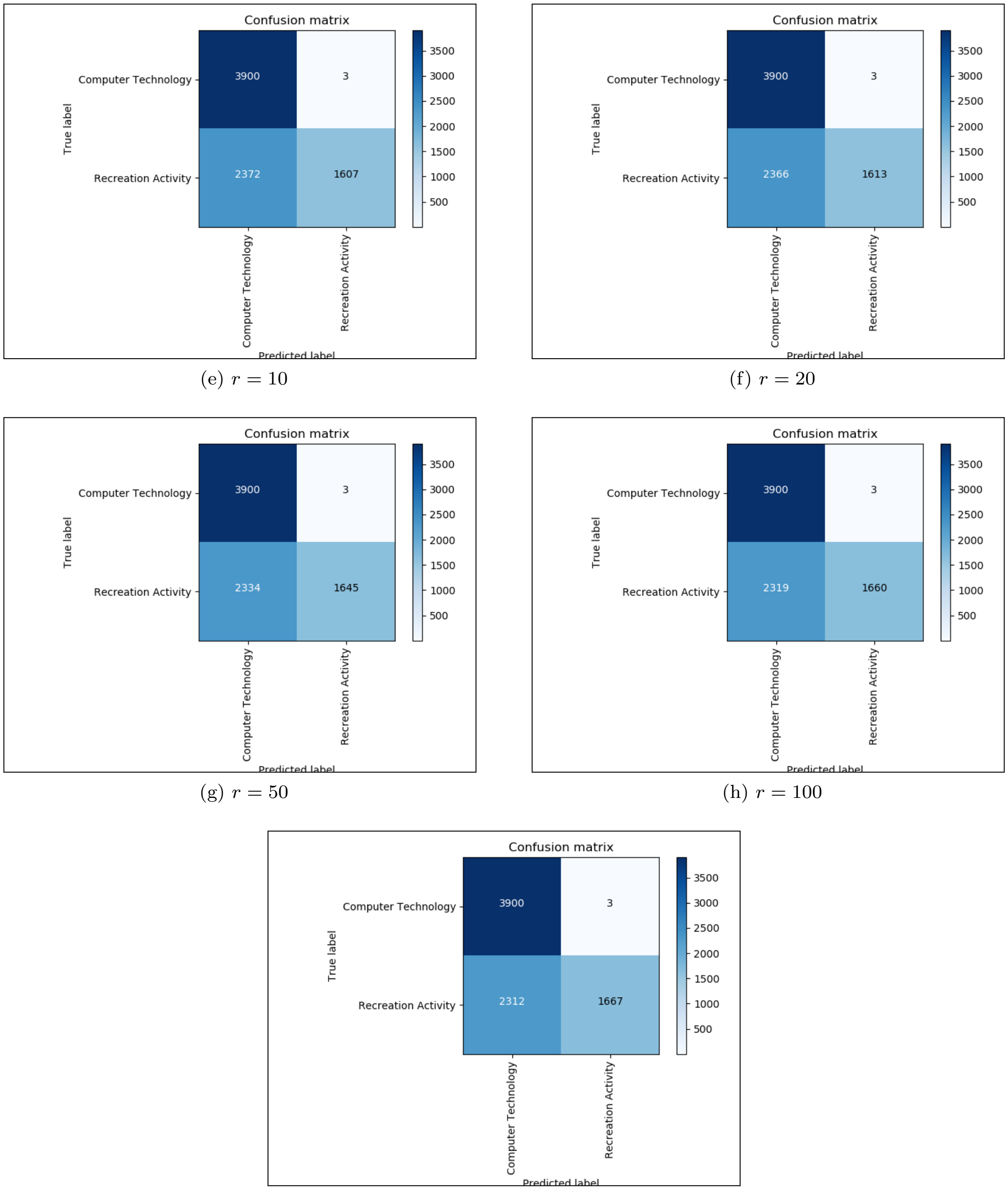
(a) $r = 1$

(b) $r = 2$

(c) $r = 3$

(d) $r = 5$

Figure 3: Contingency Matrices for Varying r



(e) $r = 10$

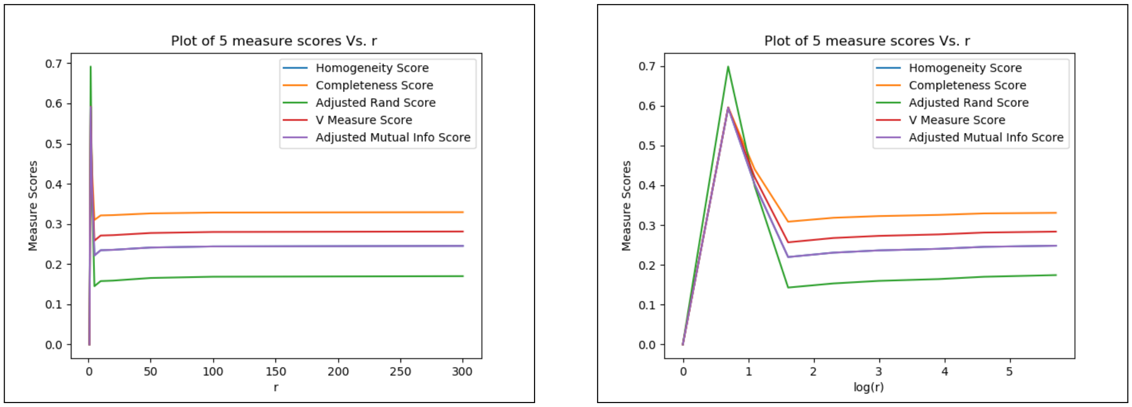
(f) $r = 20$

(g) $r = 50$

(h) $r = 100$

(i) $r = 300$

Figure 3: Contingency Matrices for Varying r



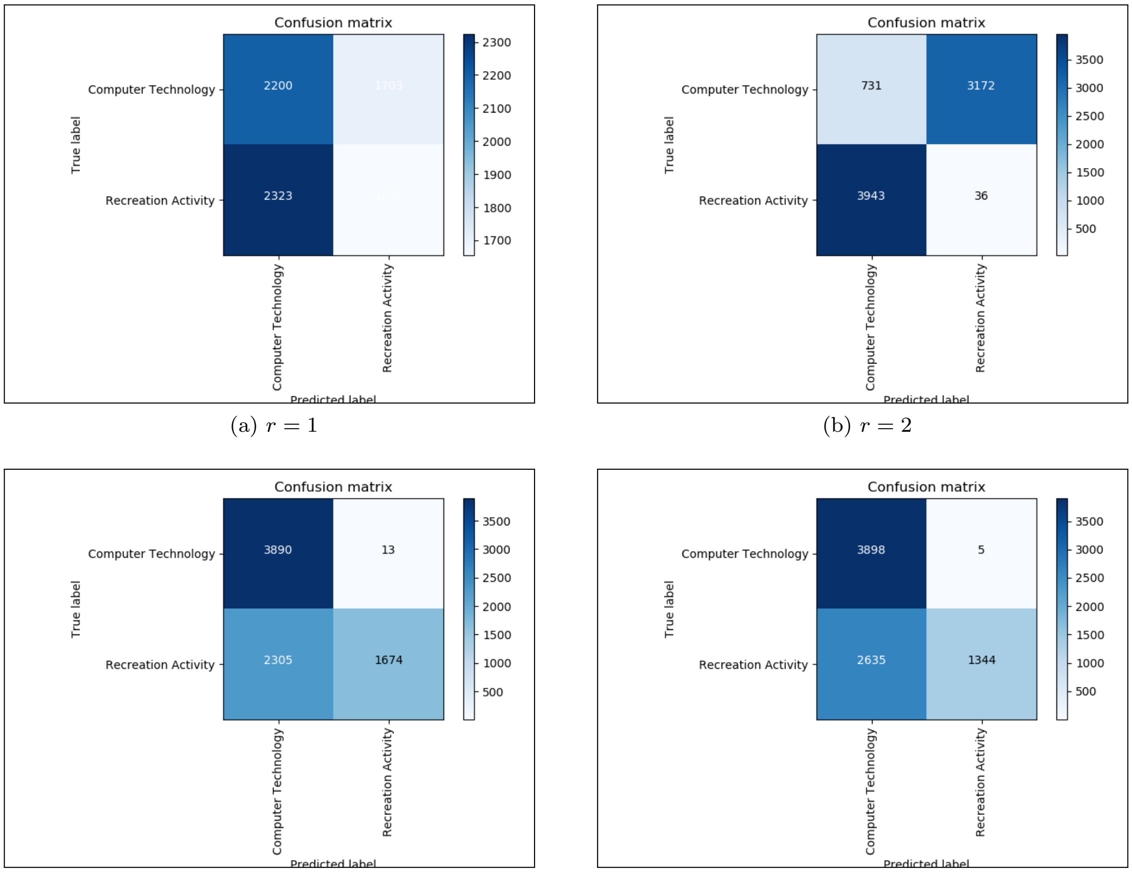
(j) Linear Scale

(k) Log Scale

Figure 4: Measure Scores

5.2.2 NMF

The contingency matrices are shown in Figure 5 and the measure scores are plotted in Figure 6.



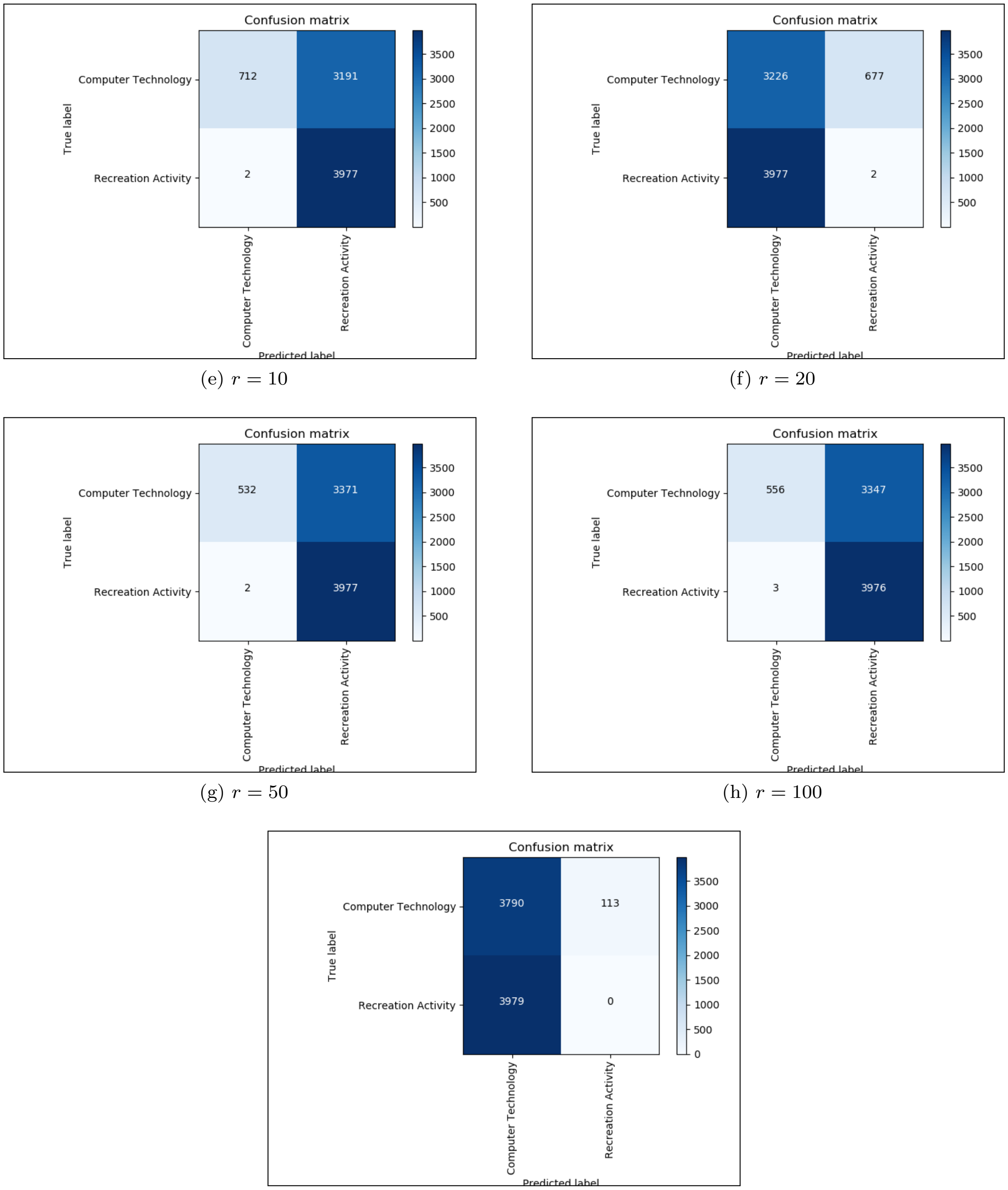
(a) $r = 1$

(b) $r = 2$

(c) $r = 3$

(d) $r = 5$

Figure 5: Contingency Matrices for Varying r



(e) $r = 10$

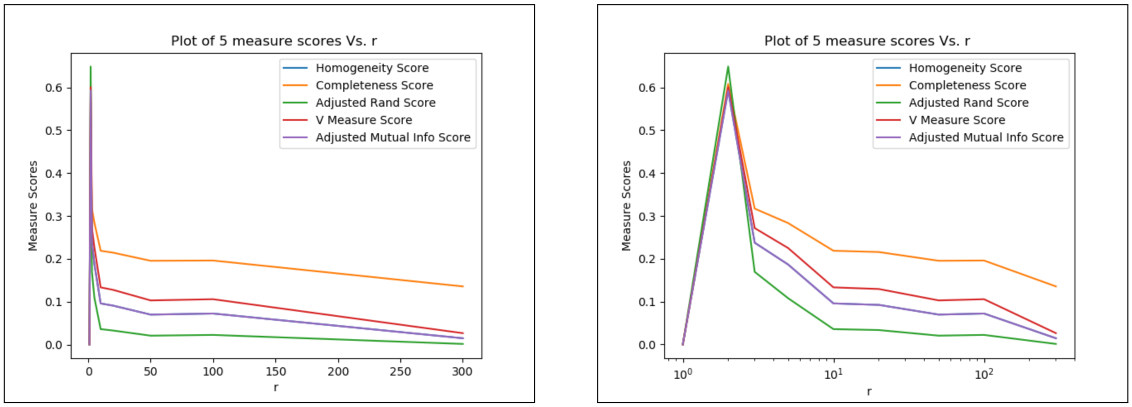
(f) $r = 20$

(g) $r = 50$

(h) $r = 100$

(i) $r = 300$

Figure 5: Contingency Matrices for Varying r



(j) Linear Scale

(k) Log Scale

Figure 6: Measure Scores

5.3 Analysis

For both LSI and NMF, $r = 2$ produced the best results across all performance metrics. The measure scores for both dimensionality reduction techniques are tabulated below.

	LSI	**NMF**
Homogeneity	0.591	0.593
Completeness	0.593	0.608
V-Measure	0.592	0.600
Adjusted Rand Index	0.692	0.649
Adjusted Mutual Information	0.591	0.593

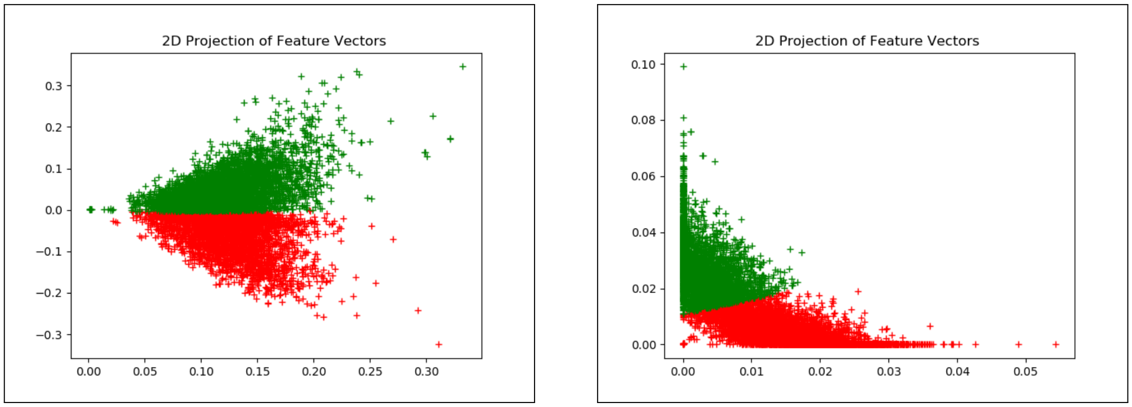
Table 2: Measure Scores for $r = 2$

As seen from Table 2, both dimensionality reduction techniques had similar homogeneity, completeness, V-Measure and adjusted mutual information, with LSI outperforming NMF most significantly for the adjusted Rand index, which measures accuracy. Therefore, LSI is the better dimensionality reduction algorithm.

6 Transformation of Feature Vectors

6.1 Projection onto 2-Dimension Plane

To visualize the spread of the dimensionality reduced data points, the feature vectors are projected onto a 2D plane, with the data points color-coded according to their respective predicted classes. This is shown in Figure 7.



(a) LSI

(b) NMF

Figure 7: 2D Projection of Feature Vectors

As expected, the projection of feature vectors by NMF only appear in the first quadrant as all elements are non-negative. It can also be observed that the elements of the feature vectors are very small, which means that they concentrate around the origin. This may not be desirable because the optimization problem would be governed by data points close to the centroid. For example, for two data points 0.01 and 0.001 away from the centroid, a lot more emphasis would be placed on the first one, as $0.01^2 \gg 0.001^2$.

6.2 Transformation of NMF Dimensionality Reduced Feature Vectors

Transformations can be carried on the feature vectors to address the problem of data points concentrating near the origin. In this section, normalization and logarithm transformation will be investigated.

Normalization of feature vectors to unit variance standardizes the spread of the distribution. For large feature values, normalization helps to ensure that each feature contributes approximately proportionately to the distance. This is because if one of the features has a broad range of values, the distance will be governed by this particular feature ($100^2 \gg 10^2$). For feature values that are small, normalization would mean a scaling factor larger than 1 is applied, hence increasing the absolute distance between data points.

The **logarithm transformation** spreads out small values and compresses large values because

$$\log x \begin{cases} \leq 1 & \text{if } x \geq 1 \\ > 1 & \text{if } 0 < x < 1 \end{cases} \tag{1}$$

Hence, this is a suitable transformation for this dataset because it helps spread out data points near the origin.

In the previous section, $r = 2$ was shown to yield to the best clustering results. Hence, the feature vectors used for the remaining of this section will be of 2 dimensions.

Normalization, logarithm transformation and their combinations in different orders are carried out on NMF dimensionality reduced feature vectors in this section. Due to the non-negative property, logarithm can be applied to the elements of the feature vectors. For normalization, the features are scaled to unit variance but not centered at zero. This is because centering at zero would yield negative elements, which logarithm cannot be applied to.

6.2.1 Normalization

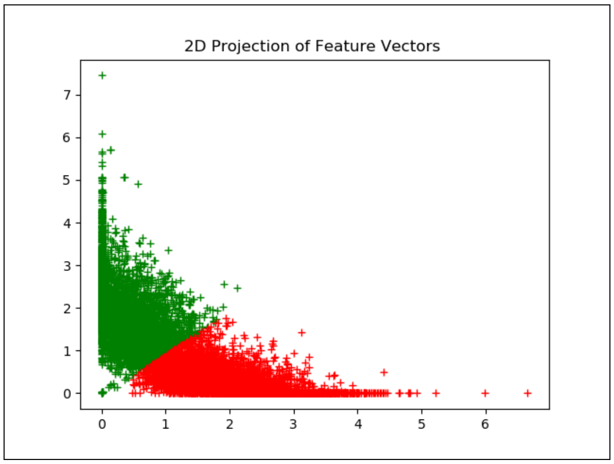


Figure 8: 2D Projection of Transformed Feature Vectors

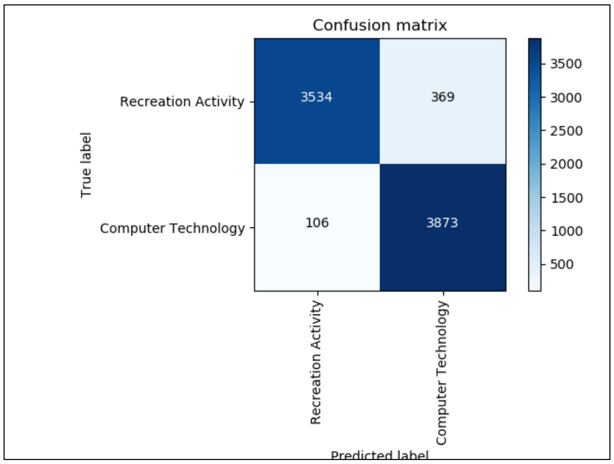


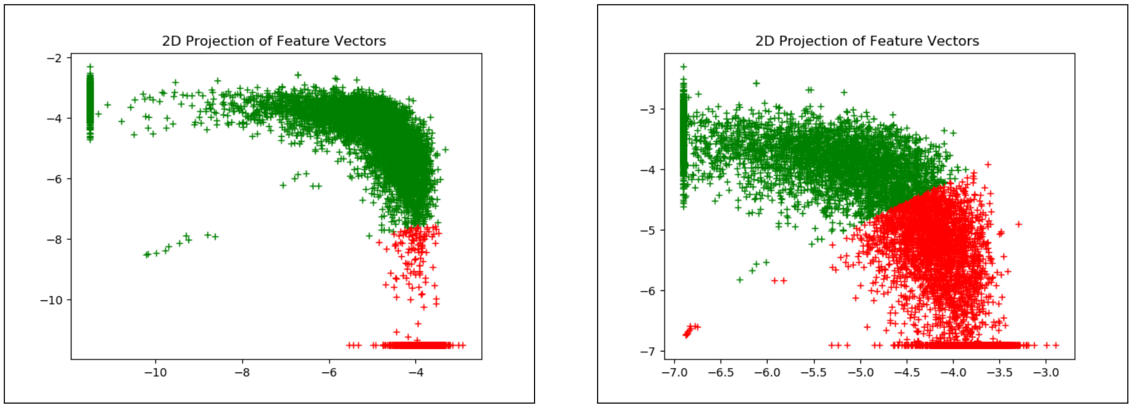
Figure 9: Contingency Matrix

Homogeneity	0.683
Completeness	0.686
V-Measure	0.684
Adjusted Rand Index	0.773
Adjusted Mutual Information	0.683

Table 3: Measure Scores

6.2.2 Logarithm Transformation

For the logarithm transform, a small constant is added to the feature vectors to ensure that none of the elements are exactly zero. When the constant is too small, the data points with zeros will appear as outliers, as shown in Figure 10(a). Hence, a suitable constant is chosen to be small with minimal outliers effect.



(a) $c = 0.00001$ (b) $c = 0.001$

Figure 10: 2D Projection of Transformed Feature Vectors

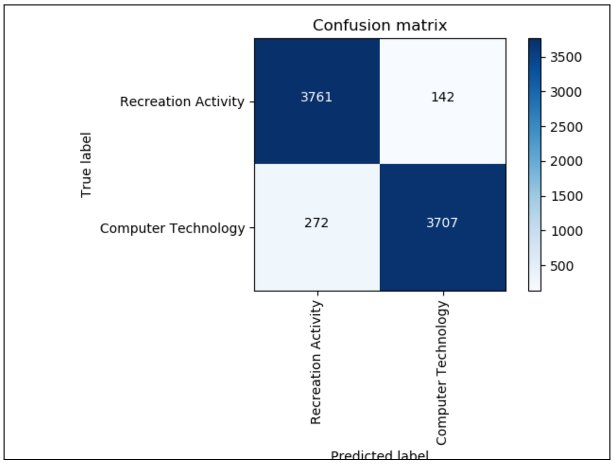


Figure 11: Contingency Matrix

Homogeneity	0.706
Completeness	0.707
V-Measure	0.707
Adjusted Rand Index	0.801
Adjusted Mutual Information	0.706

Table 4: Measure Scores

6.2.3 Normalization + Logarithm Transformation

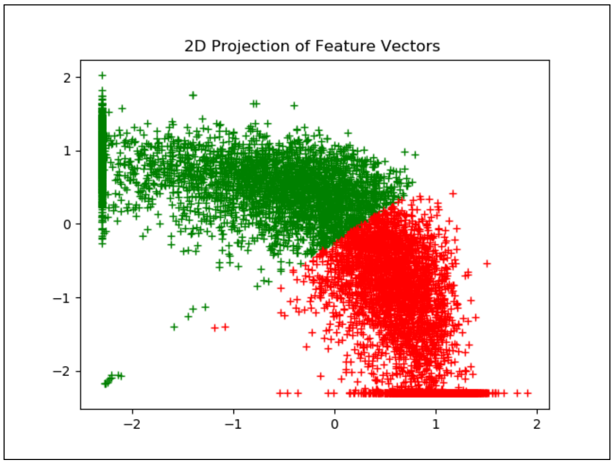


Figure 12: 2D Projection of Transformed Feature Vectors

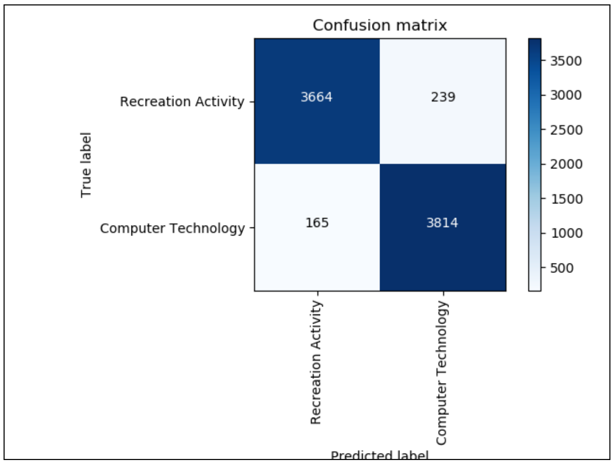


Figure 13: Contingency Matrix

Homogeneity	0.709
Completeness	0.710
V-Measure	0.709
Adjusted Rand Index	0.805
Adjusted Mutual Information	0.709

Table 5: Measure Scores

6.2.4 Logarithm Transformation + Normalization

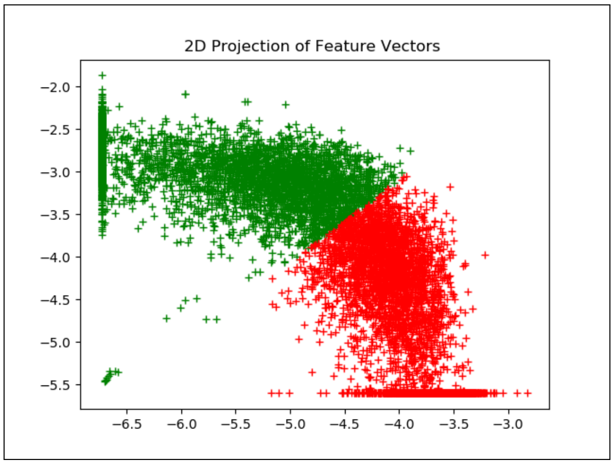


Figure 14: 2D Projection of Transformed Feature Vectors

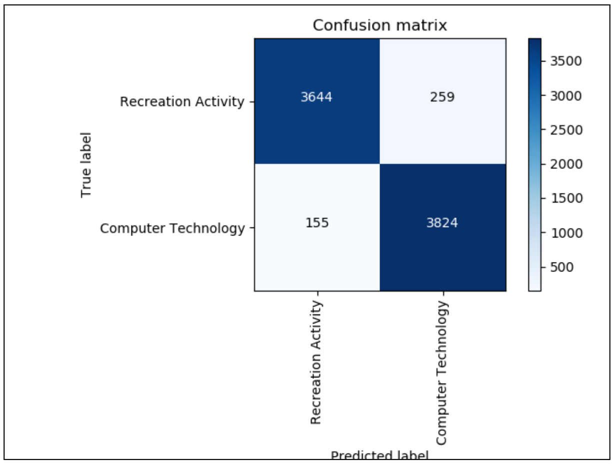


Figure 15: Contingency Matrix

Homogeneity	0.705
Completeness	0.705
V-Measure	0.705
Adjusted Rand Index	0.801
Adjusted Mutual Information	0.705

Table 6: Measure Scores

6.2.5 Analysis

Comparing the clustering results using the four different transformation, it can been observed that normalization, followed by the logarithm transformation yields the best scores for all performance metrics.

7 Extending to 20 Classes

In this section, the number of classes and clusters are extended to 20. The data set is formed using all the subclasses in the “20 Newsgroups” dataset.

7.1 Performance for Varying r

The best r is determined by sweeping different values. The values chosen are the same as before, where $r = 1, 2, 3, 5, 10, 20, 50, 100, 300$.

7.1.1 LSI

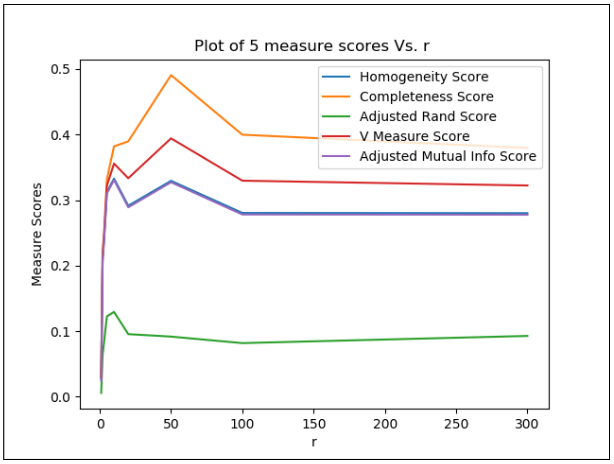


Figure 16: Measure Scores for Varying r

7.1.2 NMF

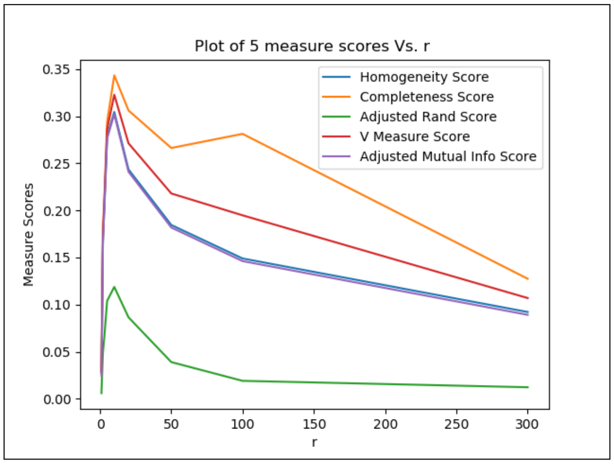


Figure 17: Measure Scores for Varying r

7.1.3 Analysis

$r = 50$ and $r = 10$ produced the best clustering results for LSI and NMF respectively. LSI performed better than NMF, which is similar to the results obtained in Project 1.

It can also be seen that the performance is worse than when k-means was carried out on just 2 clusters in the previous sections. This is probably due to strong relations between certain classes of news articles.

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Table 7: "20 Newsgroups" Dataset Classes - Grouped

Table 7 shows the potential grouping of classes which may have significant overlap, hence carrying out k-means with 6 clusters may produce better results.

7.2 Transformation of NMF Dimensionality Reduced Feature Vectors

In this section, the feature vectors are first reduced to 10 dimensions using NMF before transformation is carried out. The 2D projection of feature vectors are omitted from this section as $r = 10$ is larger than 2. Contingency matrices are also not included as it is difficult to produce the best match between 20 cluster labels and 20 ground truth labels.

Normalization, logarithm transformation and their combinations in different orders are investigated in this section. Each transformation's measure scores are tabulated below.

Homogeneity	0.301
Completeness	0.337
V-Measure	0.318
Adjusted Rand Index	0.111
Adjusted Mutual Information	0.299

Table 8: Measure Scores: Normalization

Homogeneity	0.370
Completeness	0.373
V-Measure	0.371
Adjusted Rand Index	0.201
Adjusted Mutual Information	0.368

Table 9: Measure Scores: Logarithm Transformation

Homogeneity	0.373
Completeness	0.377
V-Measure	0.375
Adjusted Rand Index	0.206
Adjusted Mutual Information	0.371

Table 10: Measure Scores: Normalization + Logarithm Transformation

Homogeneity	0.376
Completeness	0.380
V-Measure	0.378
Adjusted Rand Index	0.205
Adjusted Mutual Information	0.374

Table 11: Measure Scores: Logarithm Transformation + Normalization

Logarithm transformation, followed by normalization produced the best overall clustering results as it produced the highest measure scores for 4 out of the 5 yardsticks.