Unsupervised Learning and Dimensionality Reduction

Student Name: Sathish Sampath
GT Account Name: ssampath33

Introduction

This paper explores the different algorithms for Clustering and Dimensionality reduction and their performance on two different datasets. The pre-processed data is then used to train Artificial Neural Network and the performances are compared. The Clustering algorithms compared are K-Means and Expectation Maximization(EM). The four Dimensionality reduction techniques implemented are: Principal Component Analysis(PCA), Independent Component Analysis(ICA), Random Projection(RP) and Random Forest (feature selection algorithm of desire).

Datasets

The two datasets are QSAR biodegradation Dataset and Digits Dataset.

QSAR Biodegradation Dataset (Biodeg) – Dataset from Assignment 1

Biodegradation experimental values of 1055 chemicals are available in this dataset. The chemicals were analysed in 41 different features (Number of heavy atoms, Number of substituted benzene C, Number of nitro groups, etc.) This data is used to develop models for the study of the relationships between chemical structure and biodegradation of molecules. The Classification model developed should discriminate the chemicals as Ready Biodegradable and Not Ready Biodegradable molecules. This is an interesting problem as the biodegradability of the chemicals cannot be determined using any simple rules. The machine model developed will help us in determining the Biodegradability of the chemical more accurately. This dataset has totally 1055 instances with 41 features each. The target function is discrete valued (Biodegradable(RB) and Non-Biodegradable(NRB)).

Digits Dataset (Digits)

The Digits Dataset contains the image data of handwritten digits. The Image recognition and Optical Character Recognition(OCR) are some of the interesting applications of Machine learning. The handwritten digit image is broken into 8X8 grid and each value is present in one feature of the Digit dataset. There are approximately 180 records for each digit and 10 digits are represented in the dataset. This is an interesting dataset, as the handwriting varies from one person to another. The Digits dataset has totally 1797 instances with 64 features each. The target function is a digit (0 to 10).

Implementation

The algorithms are implemented using Scikit-learn library in Python language. The Scikit-learn is machine learning library and it features various classification, regression, clustering algorithms and dimensionality reduction algorithms. For the evaluation of the performance of the Neural Network, the overall dataset is split into two: 70% as training set and 30% as test set. For cross validation purposes, the K-Fold method with 5 folds is implemented on the training part using the Scikit-Learn's inbuilt function. The K-Fold method(for K=5) further splits the train dataset into learning dataset(80%) and cross validation dataset(20 %). For Unsupervised Learning (Clustering algorithms) and Feature Transformation techniques, the label is removed and the feature data is used directly.

Part 1 Clustering

The Clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). In this project, 2 clustering techniques, K-Means clustering and expectation maximization(EM) are explored.

K-Means Clustering

The K-Means is an unsupervised learning algorithm, in which the dataset is randomly separated into K clusters. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. In each iteration, the centroid of the cluster and the instance cluster assignment is changed until no further cluster switching is possible or the distance change is lesser than the threshold. The Euclidean Distance function is used to determine the distance between the instances and cluster centroid in the K-Means algorithm.

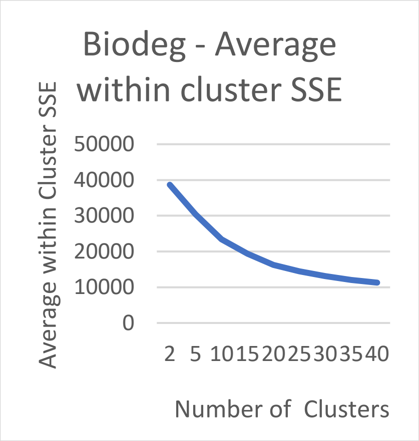
Expectation Maximization

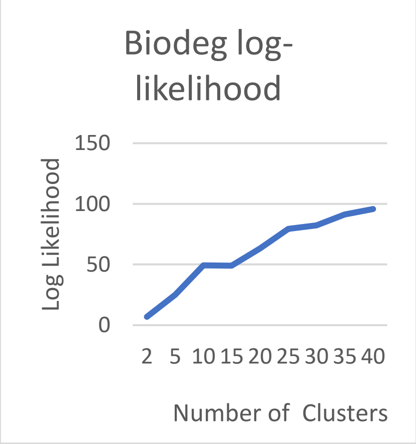
The EM algorithm finds maximum likelihood estimates of parameters in probabilistic models. EM is an iterative method which alternates between two steps, expectation (E) and maximization (M). For clustering, EM makes use of the finite Gaussian mixtures model and estimates a set of parameters iteratively until a desired convergence value is achieved. The mixture is defined as a set of K probability distributions and each distribution corresponds to one cluster. An instance is assigned with a membership probability for each cluster.

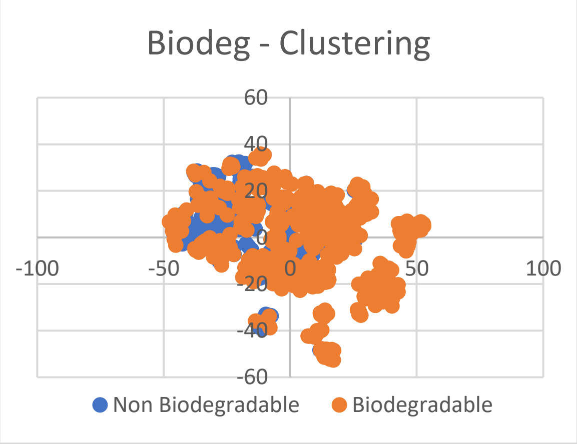
The K-Means algorithms and Expectation Maximization are implemented using the Python Scikit-learn and the performance is analysed for the different number of clusters. The comparison is done using the Accuracy score and the Adjusted Mutual Information Score. The GridSearchCV is used to compare performance for different number of cluster. The Better configuration should have higher accuracy, lower mutual information, lesser SSE score and higher log-likelihood.

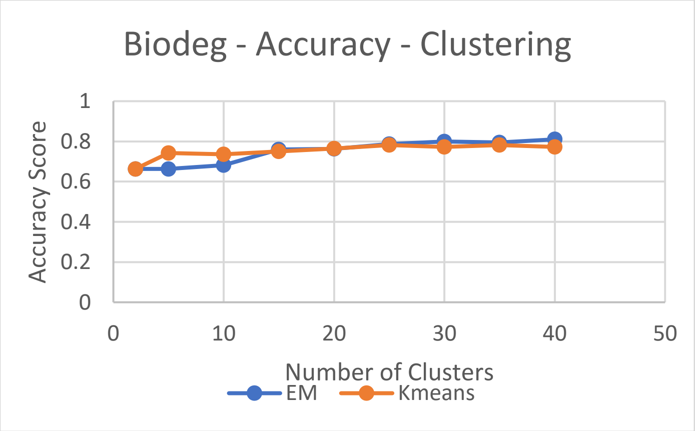
Biodeg Dataset

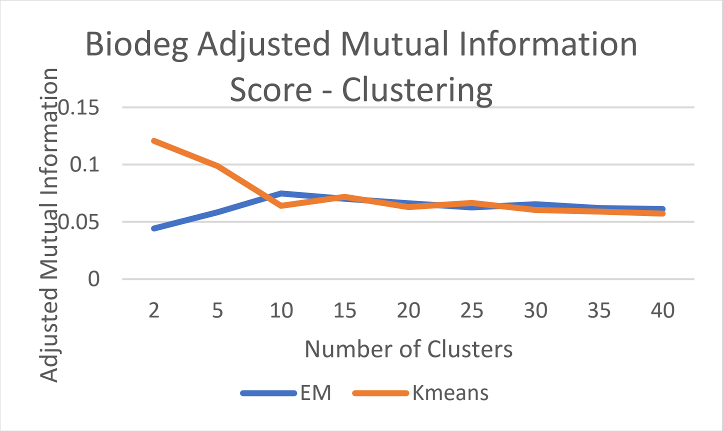
The Average within Cluster SSE score reduces and Log-Likelihood increases with the increase in Number of Clusters for the Biodeg Dataset. Compared to the K-Means, EM has lesser accuracy and lesser Adjusted Mutual Information score for lesser number of clusters, but for higher number of clusters, the EM has better accuracy. The 2D projection shows that the data instances are very close, this might be the reason for lesser overall accuracy(<0.9, for all clusters < 45). **Optimal Configuration: EM, 30 clusters with accuracy 0.81.**





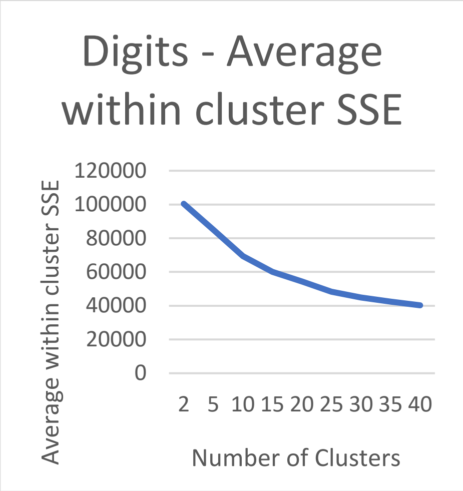


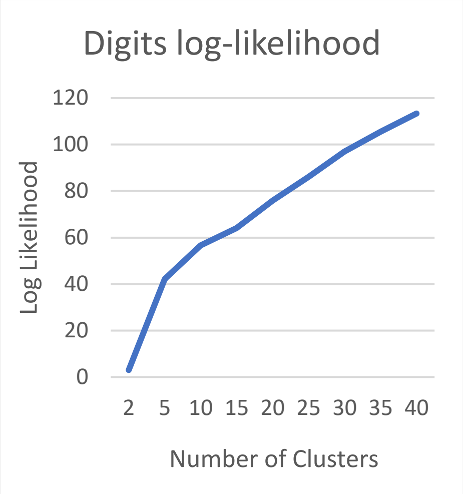


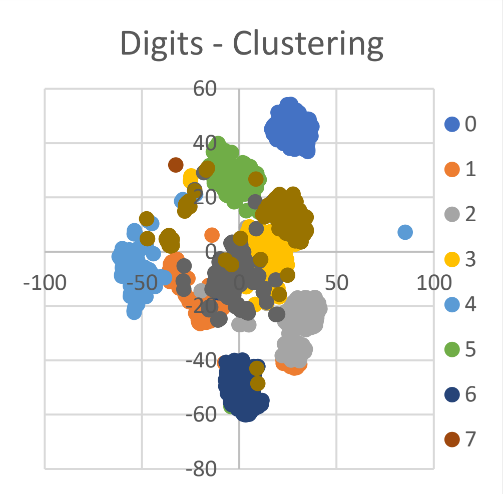


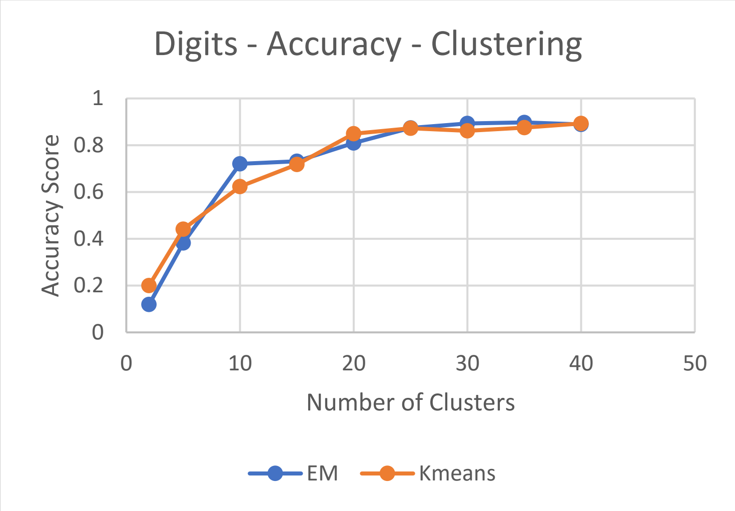
Digits Dataset

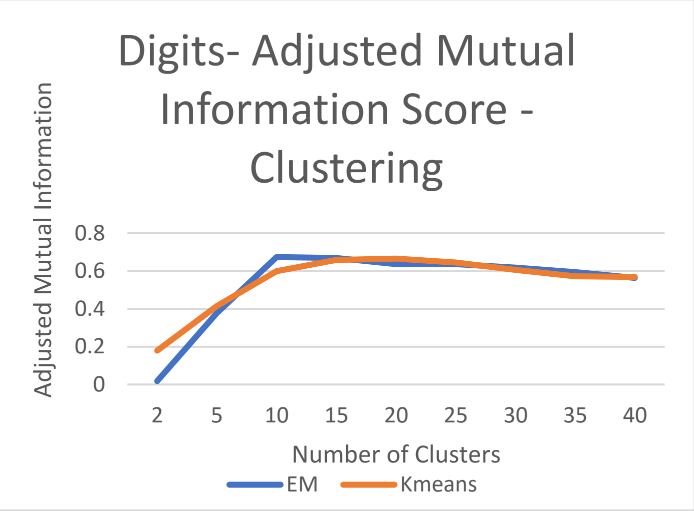
The Average within Cluster SSE score reduces and Log-Likelihood increases with the increase in Number of Clusters for the Digits Dataset. The K-Means has better accuracy and lesser Mutual Information than the EM. The 2D visualization shows that clusters are clearly separate, this might be the reason for higher accuracy score. **Optimal Configuration: K-Means, 40 clusters with accuracy 0.89,** as it has better accuracy, lower SSE, higher log-likelihood and lower Adjusted Mutual Information score.









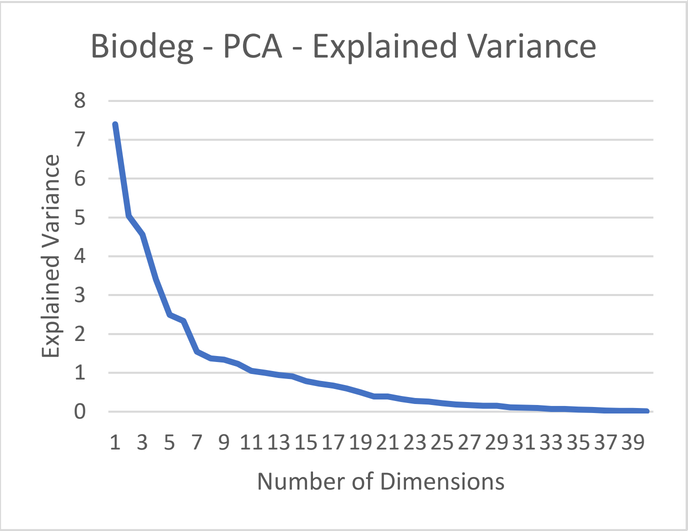


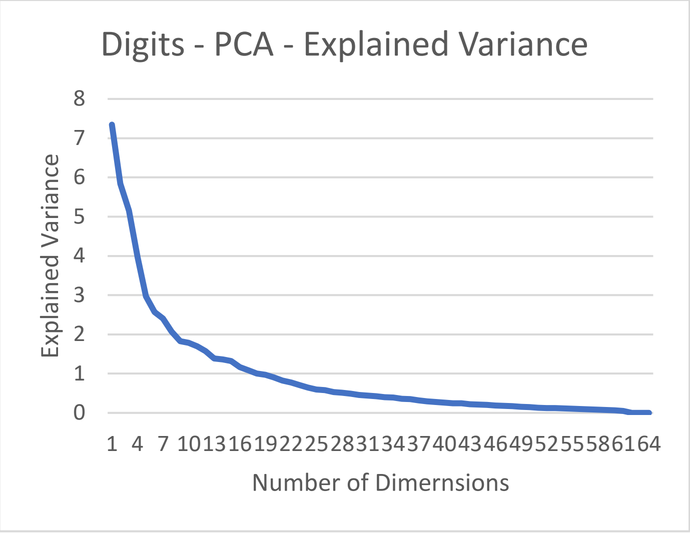
Part 2 – Dimensionality Reduction Techniques

Principal Component Analysis(PCA)

Principal Component Analysis (PCA) is a dimension-reduction technique that can be used to reduce a large set of variables to a small set that still contains most of the information in the large set. Principal component analysis (PCA) is a mathematical procedure that transforms a number of (possibly)correlated variables into a (smaller) number of uncorrelated variables called ‘principal components’. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. The eigen value for a given factor measures the variance in all the variables which is accounted for by that factor. The ratio of eigenvalues is the ratio of explanatory importance of the factors with respect to the variables. If a factor has a low eigen value, then it is contributing little to the explanation of variances in the variables and may be ignored as redundant with more important factors.

The PCA is implemented using the Python Library Scikit Learn. The PCA is applied to both the datasets and the performance is analysed. The GridSearchCV is used to compare performance for different number of dimensions. The Explained Variance is a directly related to Eigen Values, so they are used to compare the dimensions. For both the dataset, the explained variance decreases steeply for lesser values and levels down for higher values. **For Biodeg, the chosen N value is 35 and for Digits, the chosen N value is 60**, as the Explained Variance is almost similar after this value.

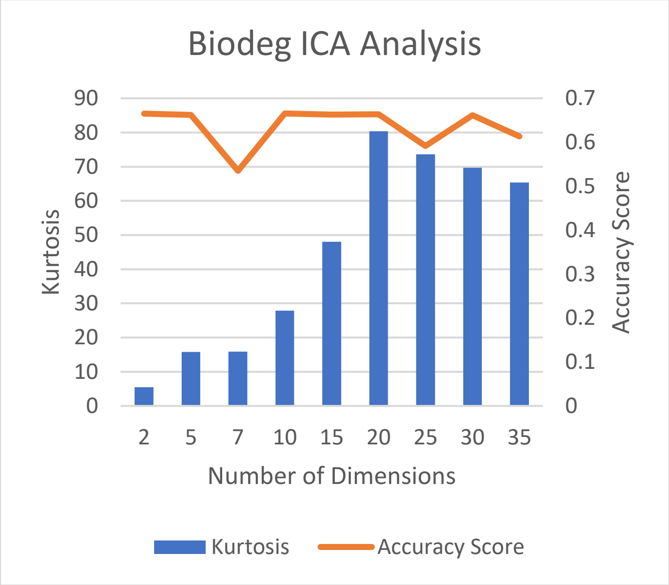


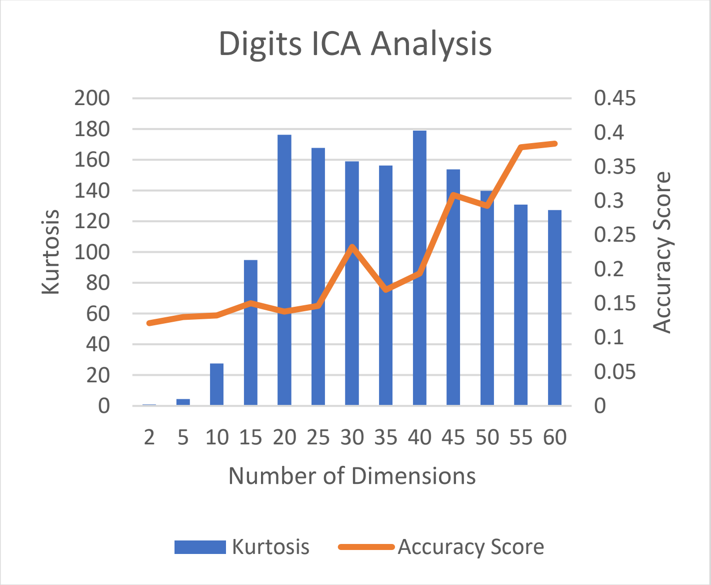


Independent Component Analysis (ICA)

Independent component analysis (ICA) is a statistical and computational technique for revealing hidden factors that underlie sets of random variables, measurements, or signals. Independent component analysis attempts to decompose a multivariate signal into independent non-Gaussian signals. In the model, the data variables are assumed to be linear mixtures of some unknown latent variables, and the mixing system is also unknown. The latent variables are assumed non-Gaussian and mutually independent, and they are called the independent components of the observed data. These independent components, also called sources or factors, can be found by ICA.

The ICA is implemented using the Python's Scikit Learn Library. The measure of non-Gaussianality, the Kurtosis value is plotted against the dimensions, along with the accuracy score.

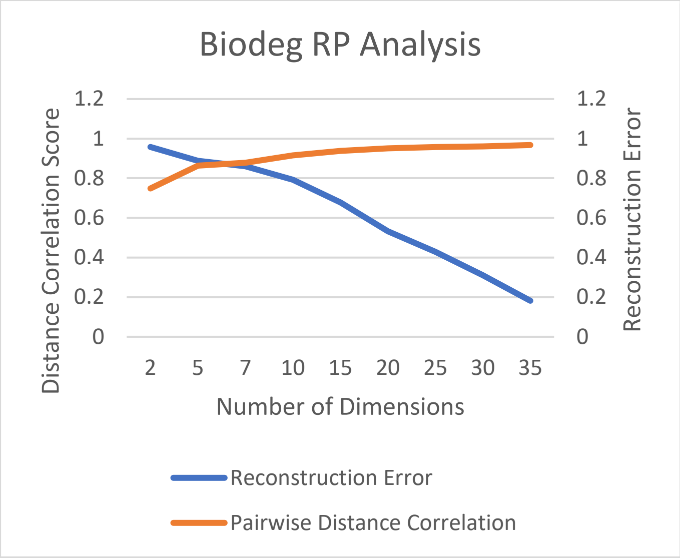


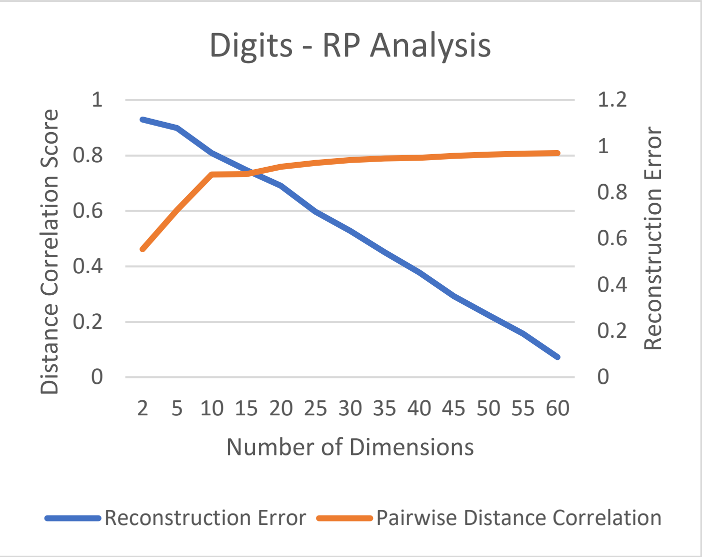


In Biodeg dataset, the Kurtosis value is low for lesser dimensions and slowly increases with the dimension. For **20 Dimensions**, there is the highest kurosis, signifying that the components are having high non- Gaussianality and are independent. The accuracy score changes very little. In the Digits dataset, the highest kurtosis value is obtained in 40 Dimensions, but the highest accuracy score is obtained in **60 dimensions**.

Random Projection(RP)

Random Projection is a technique used to reduce the dimensionality of a set of points which lie in Euclidean space. Random projection methods are powerful methods known for their simplicity and less erroneous output compared with other methods. The RP is implemented using the Python's Scikit Library. The RP algorithm is implemented for the Biodeg and the Digits datasets and the Reconstruction Error and Distance Correlation Score is plotted against the Number of Dimensions(features).



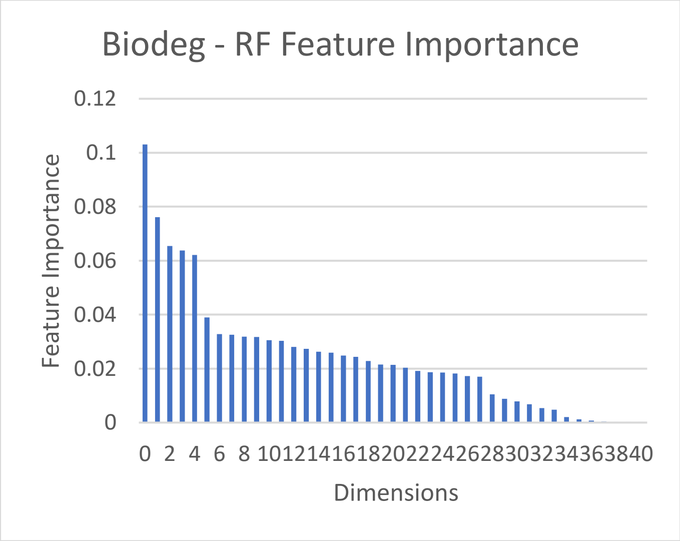


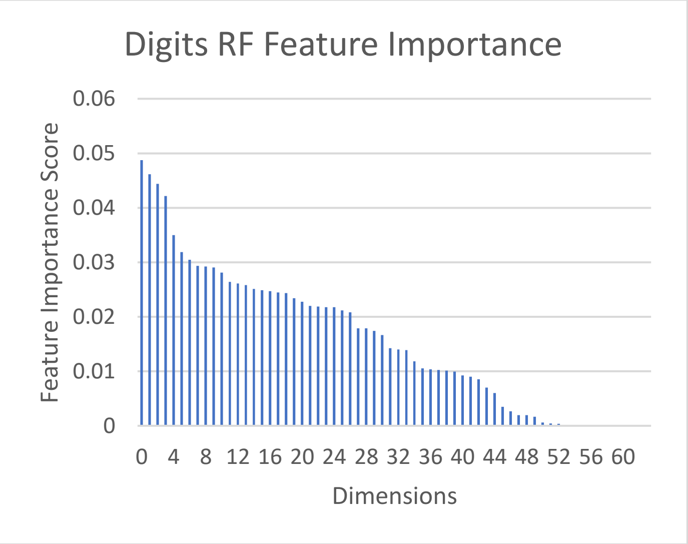
In Biodeg dataset, the Reconstruction error reduces with the increase in dimensions. The Pairwise Distance Correlation(PDC) initially increases with dimensions and then plateaus after the dimension count 15. The Reconstruction Error always decreases with increase in dimensions, so in order to determine the optimal dimension count, the PDC is used. **For Biodeg dataset, optimal Dimension = 15.** In Digits dataset, like the previous dataset, the Reconstruction error decreases with the dimensions and PDC plateaus after 60 dimensions, so **Optimal Dimension = 60, for Digits Dataset.**

Random Forest(RF)

Random forest is an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes. The Random Forest also outputs the importance of each feature which we can use to transform the data, so for the dimensionality reduction of choice, I have implemented Random Forest.

The Random Forest is implemented using the Python's Scikit Learn library for both the Biodeg dataset and the Digits dataset and the feature importance for each dimension is plotted.

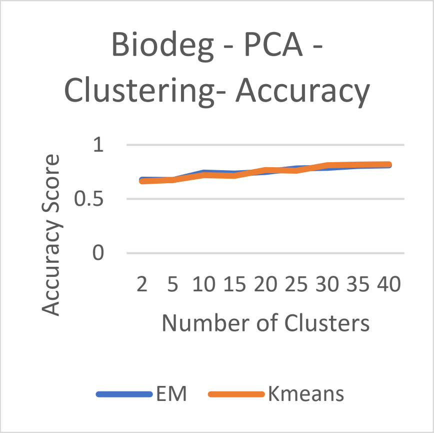


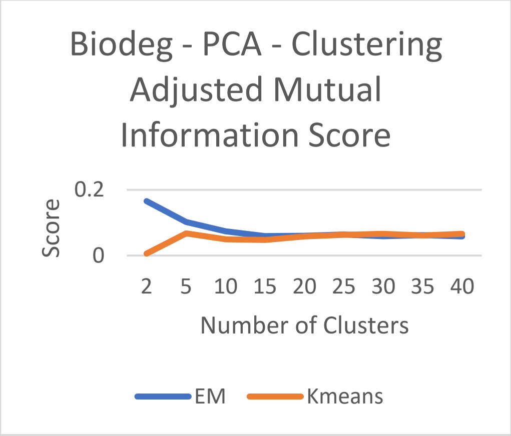


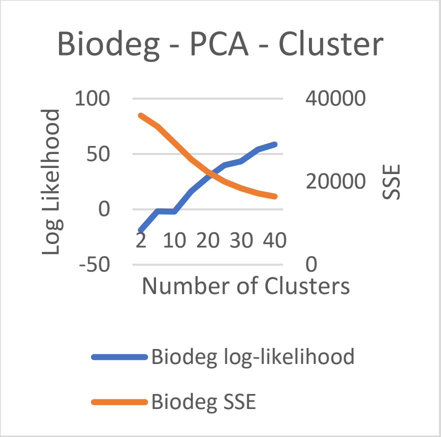
For the Biodeg dataset, the initial dimensions are having higher importance score and it reduces gradually for other dimensions. After dimension 35, for other features, the feature importance score is too low. So, the **Optimal dimension count for Biodeg dataset = 35.** For Digits dataset, the feature importance score is higher for initial features and gradually decreases. The feature importance score for dimensions after 40 is too low to consider, so the **Optimal dimension count for Digits dataset = 40**.

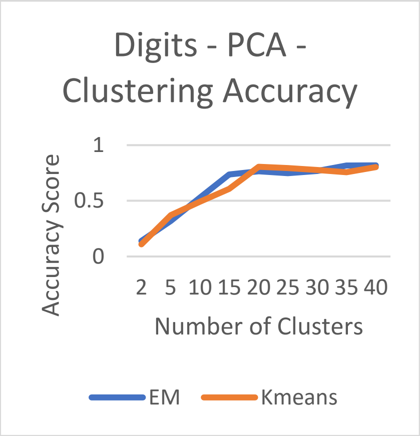
Part 3 Cluster Analysis over Dimensionality Reduction

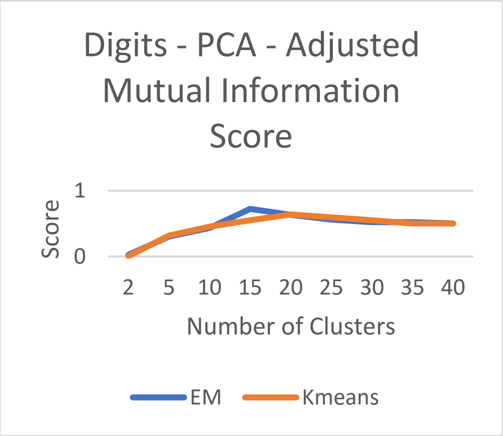
Principal Component Analysis(PCA)

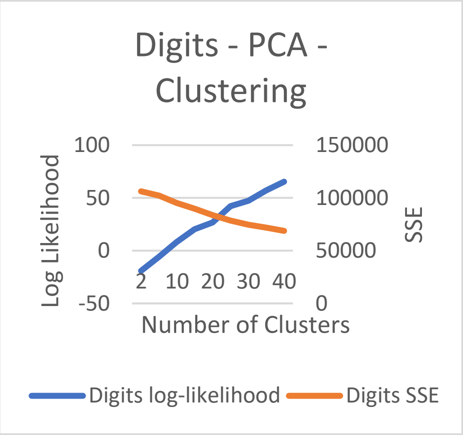






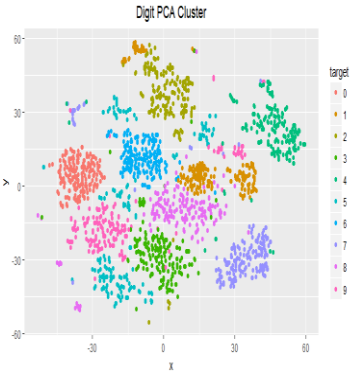


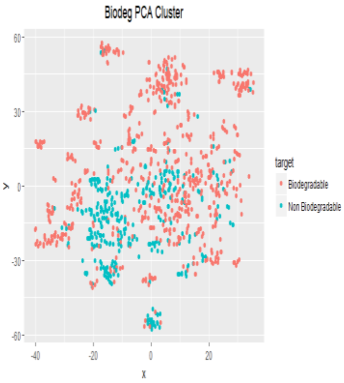


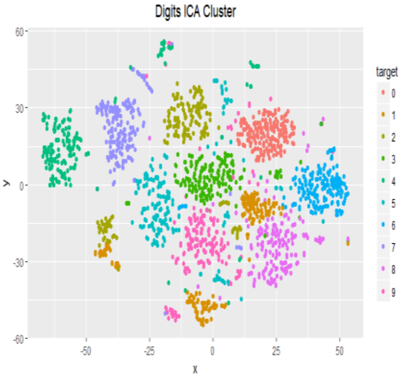


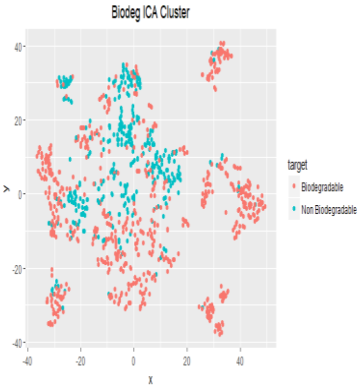
For Biodeg dataset, the cluster accuracy increases, mutual information decreases for both the clustering algorithms, as the number of clusters increase. The KMeans SSE value reduces and the EM loglikelihood increases for higher clusters and they stabilize for 40 clusters. The KMeans have better accuracy score compared to EM. The Optimal Configuration for Biodeg dataset= **K-Means, 40 clusters with accuracy 0.820683**. The Digits dataset show similar pattern, but the EM algorithm performed better in accuracy than KMeans. The SSE and Log-Likelihood graphs are smooth. Based on, the optimal configuration for Digits dataset = **EM, 40 clusters with accuracy 0.818587.**

The clusters for Biodeg dataset and Digits dataset after PCA and ICA are visualized in 2D in the following plot.





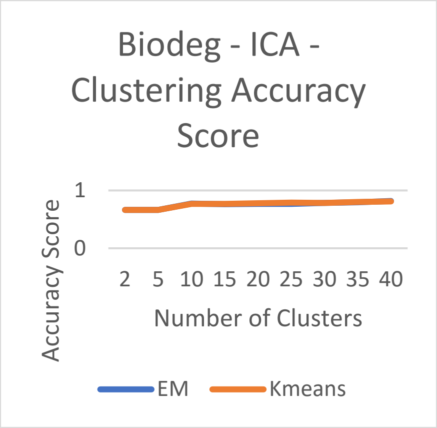


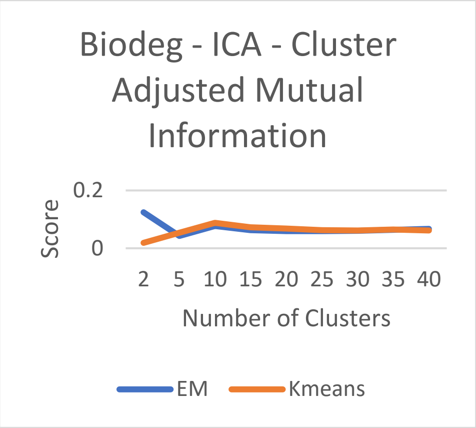


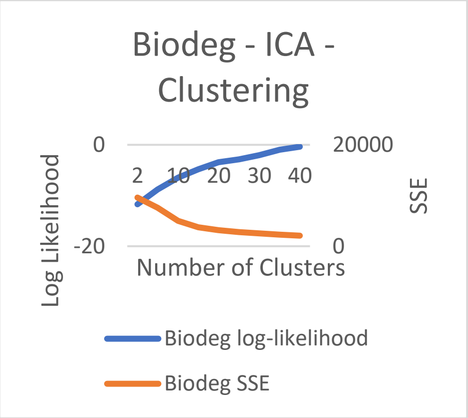
Independent Component Analysis(ICA)

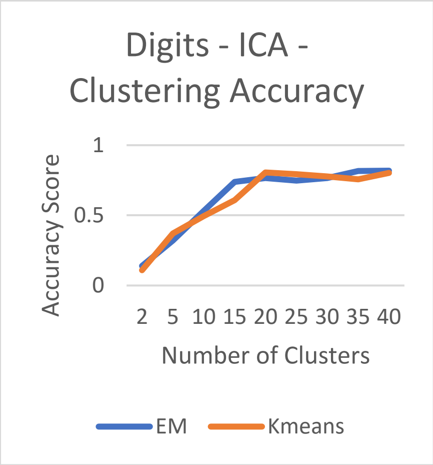
For the Biodeg dataset, after ICA implementation, the accuracy score increases with increase in number of clusters for both KMeans and EM, the KMeans being slightly better performing than EM. The KMeans also has lower Adjusted Mutual Information score, so the KMeans is selected. The KMeans SSE and EM Log-Likelihood stabilizes after 35 clusters. The Optimal configuration for Biodeg dataset is **KMeans, 35 clusters with accuracy 0.8.** For the Digits dataset, after ICA implementation, the EM is performing better than EM for higher number of clusters. The Adjusted

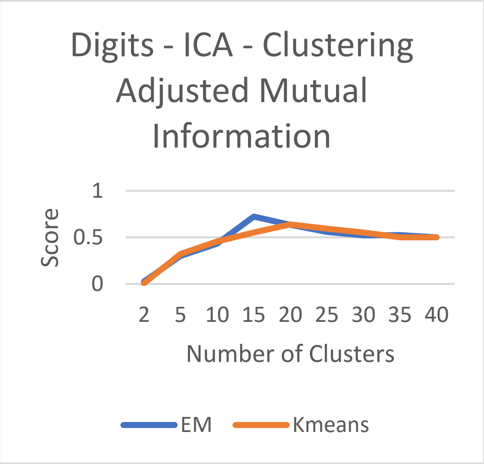
Mutual Information is also less for higher number of clusters. The KMeans SSE and EM loglikelihood graphs are smooth, so accuracy and mutual information are used to identify the optimal configuration, **EM, 40 clusters with accuracy 0.81**

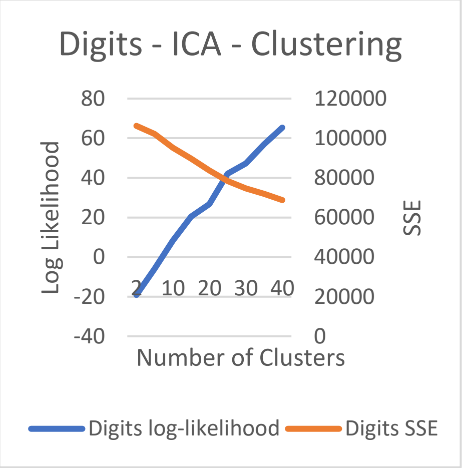




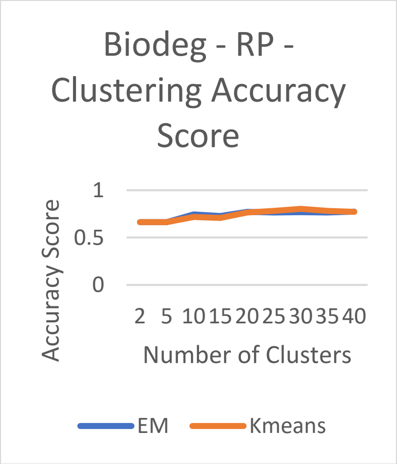


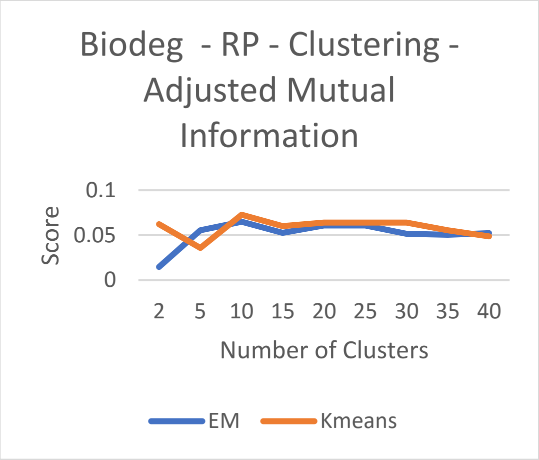


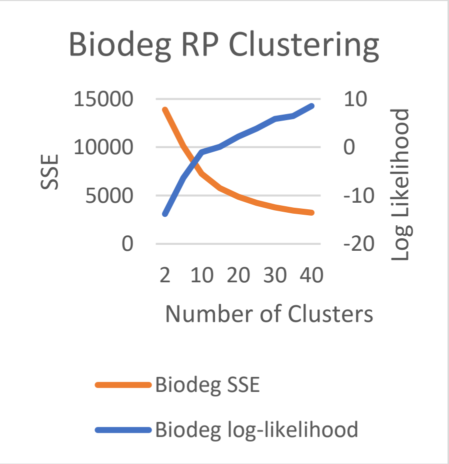


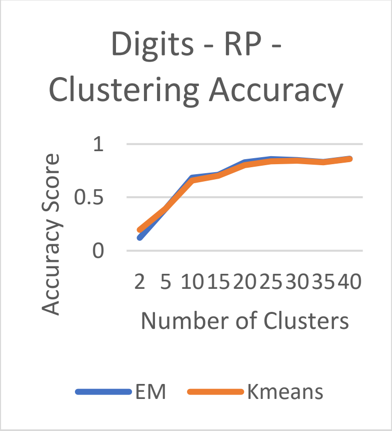


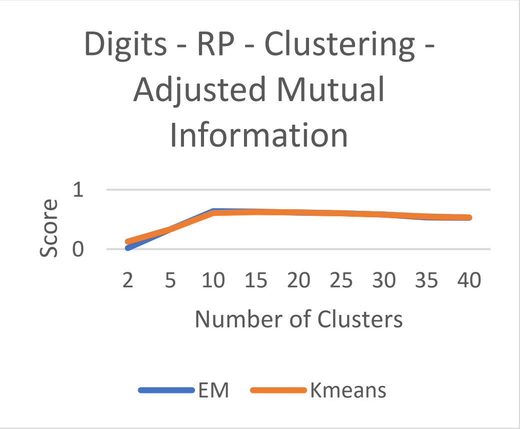
Random Projection(RP)

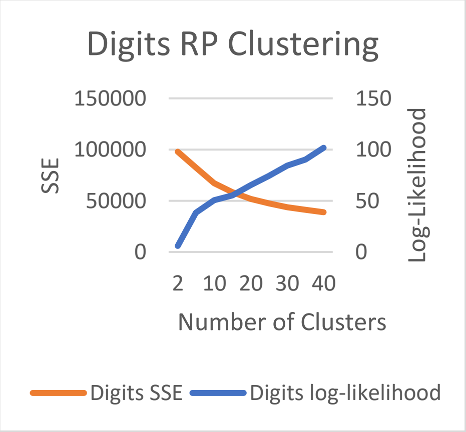






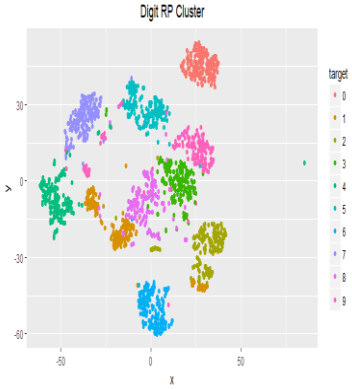


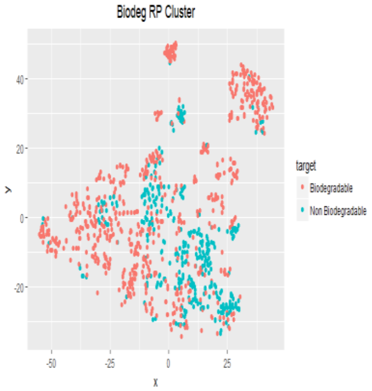


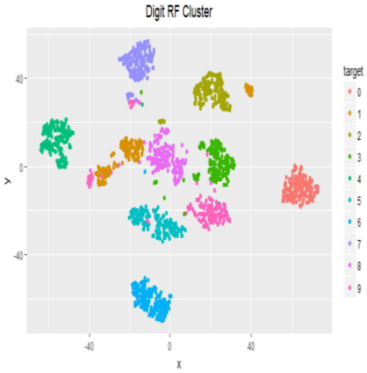


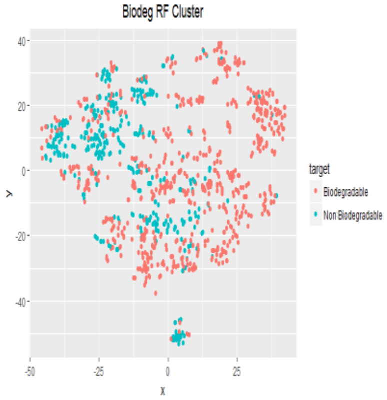
For Biodeg dataset, the SSE and Log Likelihood elbows near the 30 cluster. The KMeans is performing better for higher number of clusters compared to EM. The Optimal Configuration for the Biodeg dataset: **KMeans, 30 clusters with accuracy 0.803605.** For Digital dataset, the SSE and Loglikelihood scores had two elbow points, one at 10 clusters and another at 25 cluster. But for 10 clusters, the Adjusted Mutual Information score is higher. The EM performs better than KMeans for higher number of clusters. The Optimal Configuration for Digits dataset: **EM, 25 Clusters with accuracy 0.86.**

The Clusters for datasets after RP and RF dimensionality reduction are visualized in 2D below.



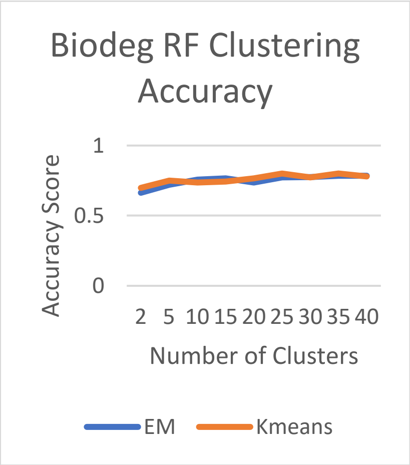


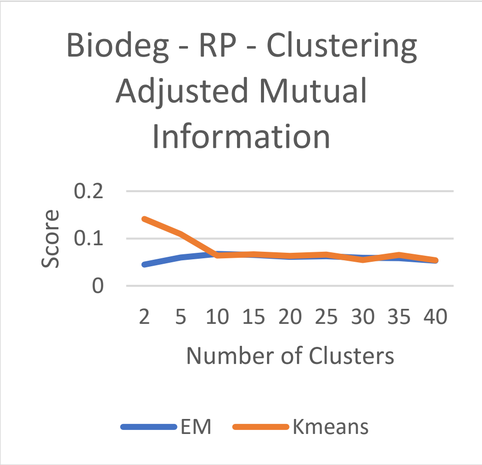


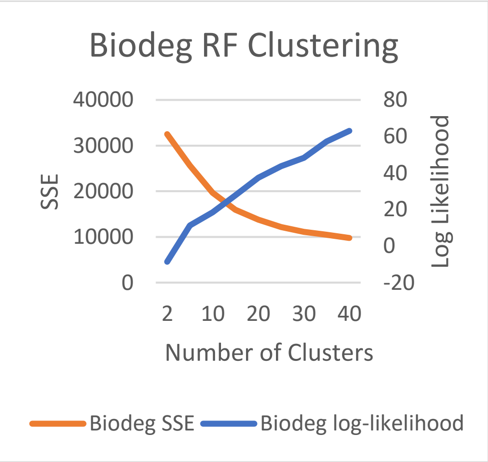


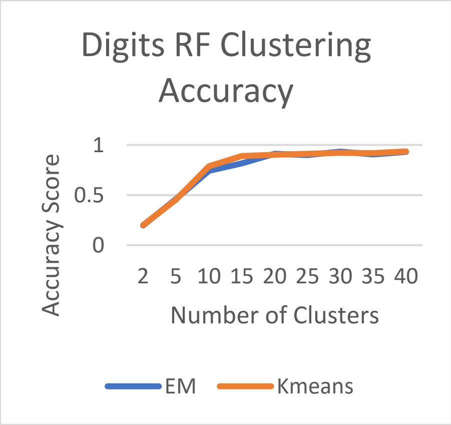
Random Forest(RF)

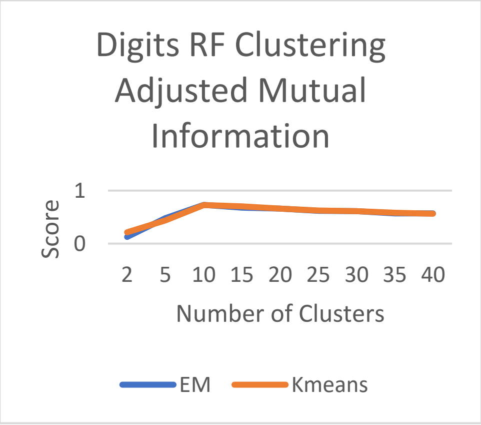
For Biodeg dataset, The SSE and Loglikelihood graphs have elbow point at 35 clusters. The KMeans has better accuracy score and lesser Adjusted Mutual Information score compared to EM for higher number of clusters. The optimal configuration for Biodeg dataset: **KMeans, 35 clusters with accuracy 0.800759.** For Digits dataset, the SSE elbow point is at 30 cluster. Even though the log-likelihood has a elbow point near 15 clusters, the Adjusted Mutual information is higher and the accuracy score is lesser, so it is not taken. The Optimal configuration for Digits dataset: **EM, 30 cluster with accuracy 0.933779.**

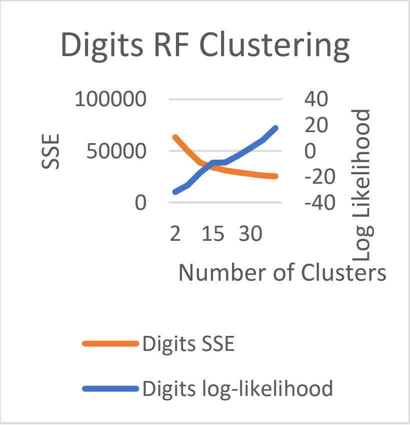






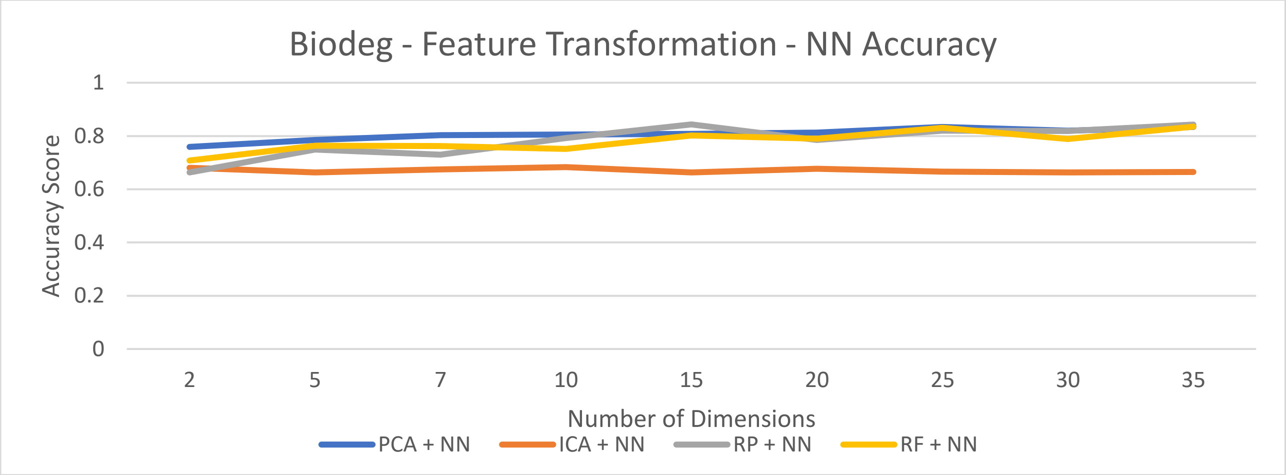






Part 4: Dimensionality Reduction with Neural Network

The Biodeg dataset after the four dimensionality reduction techniques is fed as input to the Artificial Neural Network. The ANN is implemented using the Python's Scikit Learn library. The GridSearchCV is executed to implement the Neural network in different combinations of Learning rate, Hidden Layer configuration and the Number of Dimensions. The Activation function 'Relu' is fixed, as this dataset performed well in the Assignment 1 with 'Relu' Activation.



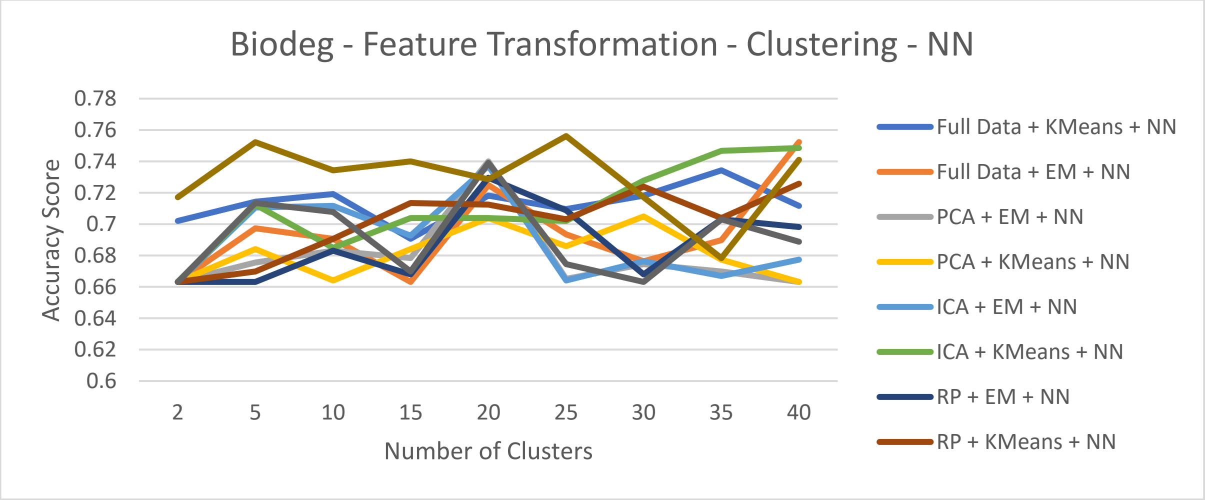
The best performance from each dimensionality reduction technique

Feature Transformation	Parameters	Accuracy Score
PCA	Learning rate 0.1, Hidden layers: (50, 50), PCA N Components: 35	0.833965844
RP	Learning rate: 0.01, Hidden layers: (100, 25, 100), RP N Components: 15	0.84345351
RF	Learning rate: 0.1, Hidden layers: (50, 50), RF Filter N dimension: 35	0.834914611
ICA	Learning rate: 0.1, Hidden layers: (50,), ICA N Components: 10	0.683111954

The NN over Biodeg dataset transformed using the Random Projection(RP) has the best Accuracy score for 15 Components and NN Parameters {Learning rate: 0.01, Hidden layers: (100,25,100)}

Part 5: Neural Network with Clustering on Dimensionally Reduced Dataset

The four dimensionality reduction techniques are applied on the Biodeg dataset. The result dataset is clustered using the two clustering algorithms – KMeans and EM and the clustering result is added as a new column to the dataset. The ANN is implemented over the resulting combined dataset and the performance is analysed.



The Dimensionality Reduction parameters(Number of Dimensions/ Components) determined from the Part 4 are directly configured for each implementation. The Neural Network with clustering is also executed on the Full dataset for comparison purposes.

The best performance from each implementation

	Parameters	**Accuracy Score**
Full data + EM + NN	Learning Rate: 0.1, Hidden Layers: (50,), Number of Clusters: 40	0.752371917
Full data + KMeans + NN	Learning Rate: 0.1, Hidden Layers: (25, 25), Number of Clusters: 35	0.734345351
PCA + EM + NN	Learning Rate: 0.1, Hidden Layers: (50,), Number of Clusters: 20	0.740037951
PCA +KMeans + NN	Learning Rate: 0.001, Hidden Layers: (100, 25, 100), Number of Clusters: 30	0.704933586
ICA + EM + NN	Learning Rate: 0.1, Hidden Layers: (50,), Number of Clusters: 20	0.737191651
ICA +KMeans + NN	Learning Rate: 0.1, Hidden Layers: (50,), Number of Clusters: 40	0.74857685
RF + EM + NN	Learning Rate: 0.1, Hidden Layers: (50,), Number of Clusters: 20	0.739089184
RF +KMeans + NN	Learning Rate: 0.1, Hidden Layers: (100, 25, 100), Number of Clusters: 25	0.756166983
RP + EM + NN	Learning Rate: 0.0001, Hidden Layers: (50,), Number of Clusters: 20	0.729601518
RP +KMeans + NN	Learning Rate: 0.1, Hidden Layers: (50,), Number of Clusters: 40	0.725806452

The NN over the Random Forest dimensionality reduced data with KMeans clustering analysis has the best performance. The NN over EM Clustering performed relatively well, as they are comparatively closer to the highest accuracy, than the other KMeans result.

Conclusion

The PCA and ICA have lower computation time and when applied clustering over it, has lower SSE and higher Log-Likelihood scores compared to other Dimensionality reduction techniques. But for Multiclass problems, the accuracy for Random Forest is much higher. From the other experimentations, the K-Means clustering worked well with binary classification problem(Biodeg) and EM clustering worked well with Multiclass classification problem(Digits). The Neural Nets when applied directly on the Dimensionally reduced dataset had better performance than the NN applied with Clustering. It is also better than the direct NN execution result – Accuracy Score 0.82. Applying Dimensionality Reduction, did increase the accuracy of the Neural Network.

Reference

1. Expectation Maximization Clustering, https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-8_289
2. Principal Component Analysis, ftp://statgen.ncsu.edu/pub/thorne/molevoclass/AtchleyOct19.pdf
3. ICA, https://www.cs.helsinki.fi/u/ahyvarin/whatisica.shtml
4. Scikit Learn, http://scikit-learn.org/
5. Code Source: Jonathan Tay(https://github.com/JonathanTay/CS-7641-assignment-3)
6. QSAR Biodegradation Dataset, UCI ML Repository, https://archive.ics.uci.edu/ml/datasets/QSAR+biodegradation