Applied Machine Learning – Assignment 4

INTRODUCTION

Unsupervised learning is a class of machine learning techniques to find the patterns in data. The data given to unsupervised algorithm are not labelled, which means only the input variables are given with no corresponding output variables. The most common unsupervised learning method is cluster analysis, which is used for exploratory data analysis to find hidden patterns or grouping in data. In this assignment I have experimented with two clustering algorithms i.e. k-means and expectation maximization on the two datasets that I used for the previous assignments.

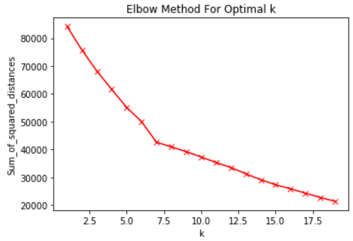
In addition, I have also implemented and experimented with several dimensionality reduction algorithms such as Decision trees, PCA, ICA and Randomized projections. Dimension reduction refers to the process of converting a set of data having vast dimensions into data with lesser dimensions ensuring that it conveys similar information concisely.

- **Dataset1 –** The first dataset used in this assignment is the Facebook comment prediction dataset.
- **Dataset2 –** The second dataset used is the Admission Prediction dataset.

Dataset 1(Facebook Comment Prediction Dataset)

Performed exploratory data analysis to find the relationship between different features and to understand the distribution of features. We found that range of values for different features differ a lot, so I performed feature scaling on these features. Performed outlier's detection and imputation. ***(*Detailed analysis done in the previous assignments)***

K Means Implementation – The K means algorithm finds the clusters and data set labels for a pre-chosen K. To find the number of clusters in the data, we need to run the K-means clustering algorithm for a range of K values and compare the results. In general, there is no method for determining exact value of K, but an accurate estimate can be obtained using the elbow method.



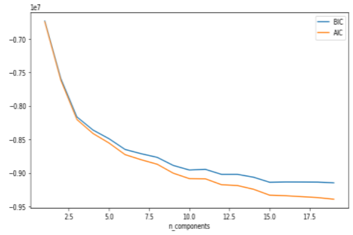
The figure shows the output of the elbow method. It shows that we should use 7 number clusters for the K means implementation.

Calculated the score for some of the metrics to check the cluster performance-

Metrics	Score	Interpretation
Adjusted Rand Index	2.78E-05	Random Label assignment have an ARI score close to 0.0
		The clusters are not similar
Adjusted Mutual Information	9.40E-06	Random Label assignment have an AMI score close to 0.0
		Values close to zero indicate two label assignments that are largely independent
Homogeneity	1.30E-04	Each cluster does not contain only members of a single class, as the score is almost 0
Completeness	4.62E-05	All members of a given class are not assigned to the same cluster, as the score is almost 0
Silhouette	0.324	Not so dense clusters, scores around zero indicate overlapping clusters

(Note: K-means clustering technique assumes that we deal with spherical clusters and each cluster has equal numbers for observations. The spherical assumptions must be satisfied. The algorithm can't work with clusters of unusual size.)

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 figure shows the output of AIC/BIC for different number of components. It shows that we should use 15 number clusters for the EM implementation according to both AIC and BIC criteria.

Calculated the score for some of the metrics to check the cluster performance-

Metrics	Score	Interpretation
Adjusted Rand Index	8.90E-03	Random Label assignment have an ARI score close to 0.0
		The clusters are not similar
Adjusted Mutual Information	1.51E-02	Random Label assignment have an AMI score close to 0.0
		Values close to zero indicate two label assignments that are largely independent
Homogeneity	5.23E-02	Each cluster does not contain only members of a single class, as the score is almost 0
Completeness	1.51E-02	All members of a given class are not assigned to the same cluster, as the score is almost 0
Silhouette	3.58E-01	Not so dense clusters, scores around zero indicate overlapping clusters
		The score is higher when clusters are dense and well separated

Feature Dimensionality Reduction algorithms

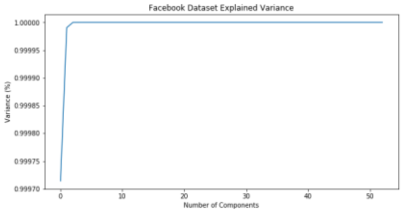
From the above results we can conclude that EM clustering provide more similar clusters to the class labels as compared to K-Means clustering. EM gives a greater number of clusters as compared to K-means.

Decision Tree Implementation- This is a filtering technique for feature selection**.** We used the same hyperparameters as used in the previous assignment and found the feature importance to perform feature selection.

Top Features	Feature Importance
AverageCC5	0.015
MedianCC1	0.017
CC1	0.018
Pagetalking about	0.019
PagePopularitylikes	0.022
CC4	0.026
CC5	0.027
PostShareCount	0.048
Postlength	0.067
Basetime	0.102
CC2	0.382

We can use these top features to build our models for further analysis to get improved results.

PCA Implementation – This is a feature transformation technique for dimensionality reduction.



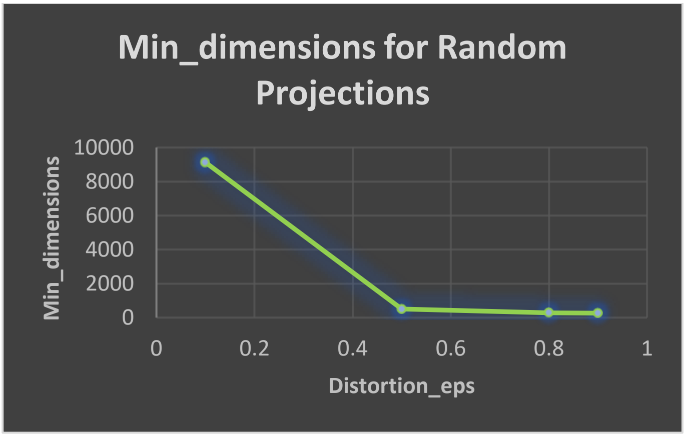
The figure shows that 2 components will give maximum cumulative variance %. So, I will use 2 components for PCA for further analysis on this dataset.

```
original shape:  (41949, 53)
transformed shape:  (41949, 2)
```

ICA Implementation- Maximizing independence between components is closely related to maximizing their non-Gaussian. In principle, this can be achieved by maximizing the absolute or squared kurtosis, which is one way to measure non-Gaussian. But, estimating kurtosis is highly sensitive to outliers, so this doesn't provide a good objective function for ICA in practice. The standard ICA model assumes the same number of sources as input dimensions, so it doesn't really provide a choice. One common way to fit fewer components is to reduce the dimensionality using PCA before running ICA. But, reducing the dimensionality too far may result in ICA finding components that are mixtures of the true sources. A subset of 'meaningful' components could then be identified using domain knowledge about the signals (by examining the component weights and time courses of the extracted signals) and/or using a model of the physical system that generated the data. So, for this assignment I will be using the number of reduced features obtained from PCA.

```
original shape: (41949, 53)
transformed shape: (41949, 2)
```

RP implementation-In RP, a higher dimensional data is projected onto a lower-dimensional subspace using a random matrix whose columns have unit length**.** The key idea of random mapping arises from **Johnson-Lindenstrauss Lemma.**

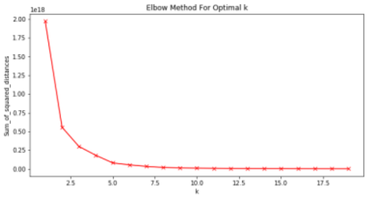


The minimum dimensions according to **Johnson-Lindenstrauss Lemma** for this dataset is 510 for distortion eps = 0.5, As our dimensions is 53. So, I will take 53 as the number of components for RP. Basically, by reducing the number of components RP's performance will decrease and will not help in dimensionality reduction. We can try finding minimum dimensions by reducing the sample size, but this will not be a good option.

```
original shape: (41949, 53)
transformed shape: (41949, 53)
```

(Note: Can use same number of clusters as decided by PCA for easy implementation or can implement Randomized PCA)

K means implementation after applying dimensionality reduction using PCA –

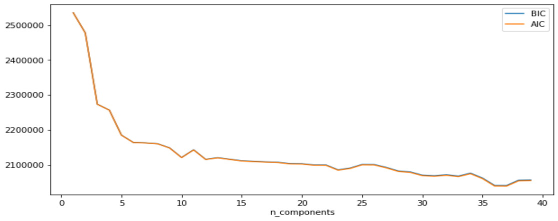


The figure shows the output of the elbow method. It shows that we should use 2 number clusters for the K means implementation.

Metrics	Score
Adjusted Rand Index	-5E-05
Adjusted Mutual Information	0.00011
Homogeneity	0.00013
Completeness	0.06129
Silhouette	0.35755

We can see from the table that ARI score is negative which shows independent labeling. The other scores are almost similar as before. The cluster have become a little denser as the silhouette score has increased a bit

EM implementation after applying dimensionality reduction using PCA –

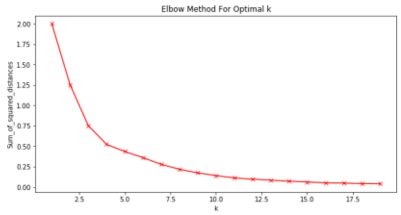


The figure shows the output of AIC/BIC for different number of components. It shows that we should use around 20 number clusters for the EM implementation according to both AIC and BIC criteria.

Metrics	Score
Adjusted Rand Index	0.0133
Adjusted Mutual Information	0.02326
Homogeneity	0.09899
Completeness	0.02333
Silhouette	0.4914

We can see from the table the scores have improved when we have used the data after using PCA, the clusters are now somewhat similar to the class labels. The clusters are comparatively denser as compared to before.

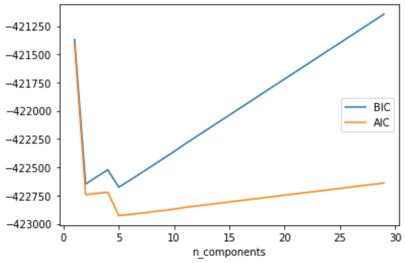
K means implementation after applying dimensionality reduction using ICA –



The figure shows the output of the elbow method. It shows that we should use 4 number clusters for the K means implementation.

Metrics	Score	Interpretations
Homogeneity	**0.02251**	Increased a bit, but still as the score is close to zero, so each cluster does not contain only members of a single class
Silhouette	**0.305**	Not so dense clusters

EM implementation after applying dimensionality reduction using ICA –

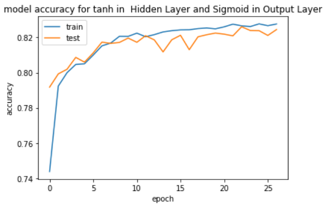


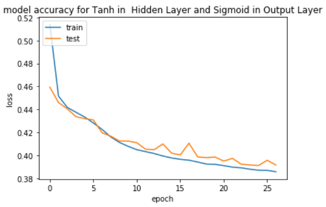
The figure shows the output of AIC/BIC for different number of components. It shows that we should use around 5 number clusters for the EM implementation according to both AIC and BIC criteria.

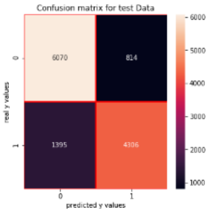
Metrics	Score	Interpretations
Homogeneity	**0.00013**	The score is close to zero, so each cluster does not contain only members of a single class
Silhouette	**0.3500**	Not so dense clusters

Neural Network Implementation-

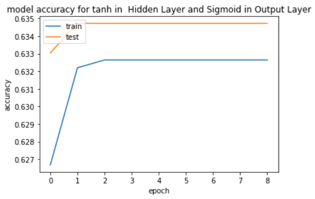
Model without dimension reduction- Used tanh activation function in the hidden and sigmoid in the output layers. loss: 0.3857 - acc : 0.8276 - val_loss: 0.3916 - val_acc: 0.8245

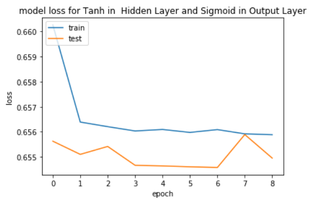


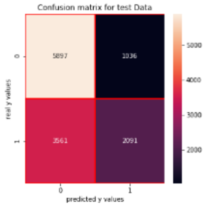




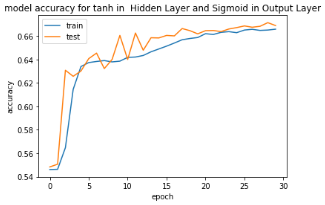
Model results after using PCA for dimensionality reduction- loss: 0.6559 - acc: 0.6326 - val_loss: 0.6550 - val_acc: 0.6347

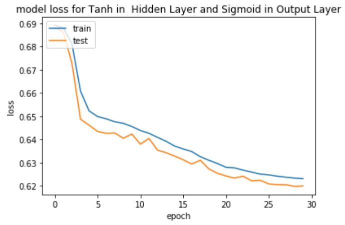


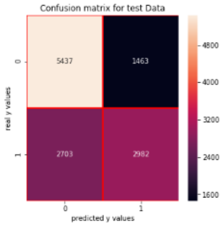




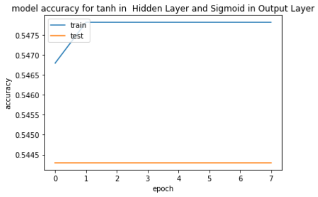
Model results after using ICA for dimensionality reduction– loss: 0.6231 - acc: 0.6659 - val_loss: 0.6200 - val_acc: 0.6690

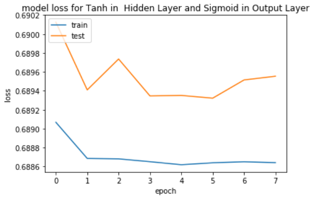


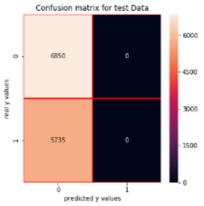




Model results after Using the clustering results from task 1 as the new features and apply neural network learner on this new data consisting of only clustering results as features and class label as the output. – loss: 0.6886 - acc: 0.5478 - val_loss: 0.6896 - val_acc: 0.5443



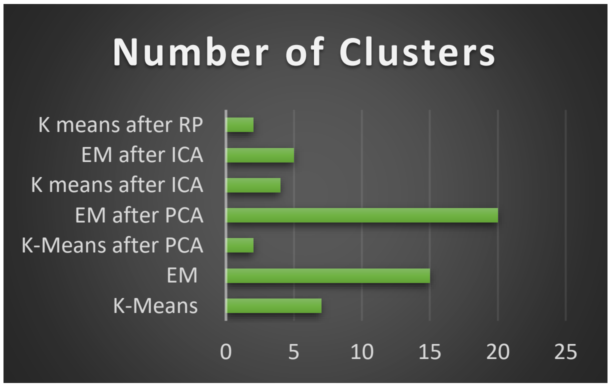


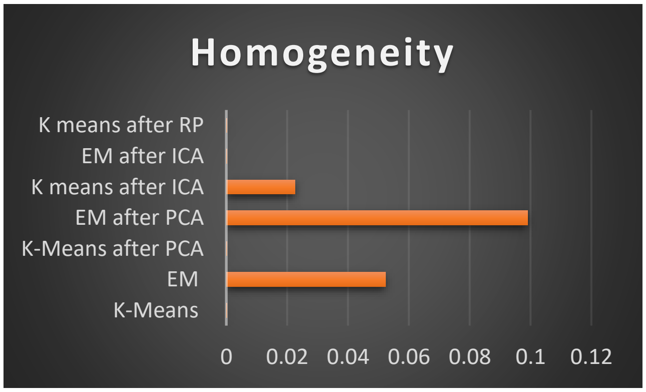


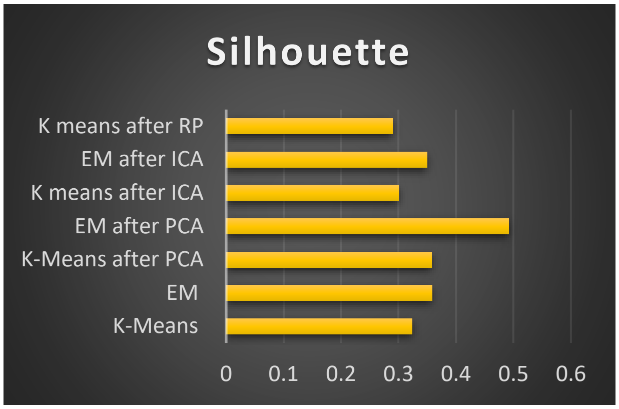
Comparission of Models – The below graphs shows the comparassion of different models that I have experimented with in this assignment for Facebook dataset.

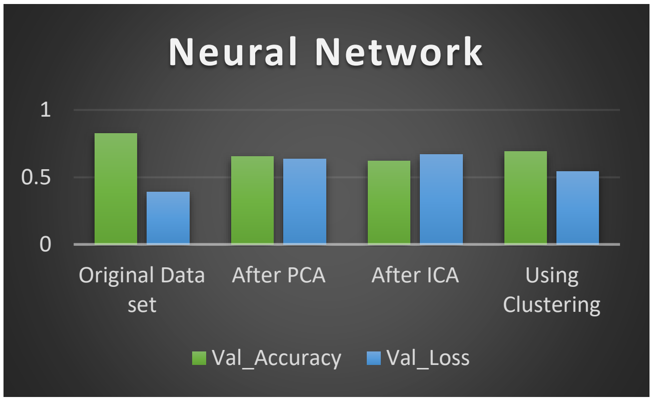
- Got different number of clusters for different experiments that were conducted
- The homogeneity is less for all the variations of models which shows that points of different nature are also present in the same cluster(very less similarity with the original class lables)
- The Silhouette score ranges from 0.3 to 0.5 which shows that the cluster are not so dense
- After applying dimensionality reduction on this data set the accracy of the neural network model decreased for all the variations
- We can also compare the shapes of the final dataset obtained depending on the number of clusters to have a better visualization of the data

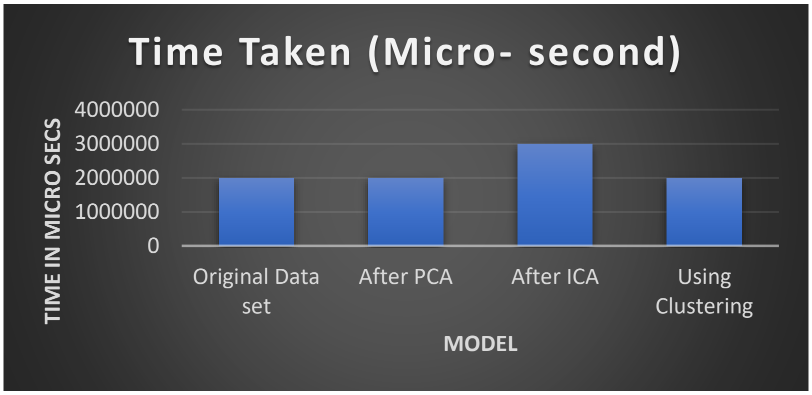
> *On an average accuracy of the neural network decreased after using different dimensionality reduction methods. Speed does not change much*







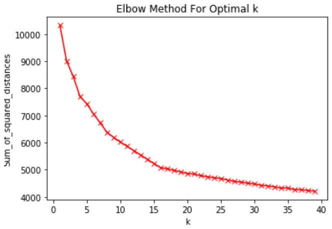




Dataset 2 (Admission Prediction dataset)

Performed exploratory data analysis to find correlation among features. Performed feature scaling and outlier detection. ***(*Detailed analysis done in the previous assignments)***

K Means Implementation –

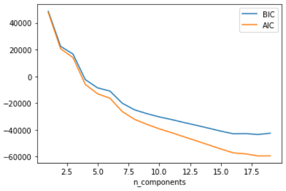


The figure shows the output of the elbow method. It shows that we should use 16 number clusters for the K means implementation.

Metrics	Score
Adjusted Rand Index	0.00242
Adjusted Mutual Information	0.00358
Homogeneity	0.01652
Completeness	0.0041
Silhouette	0.18886

We can see from the table that the clusters are not similar to the class labels. As per the Silhouette value the clusters are not dense.

Expectation Maximization-



The figure shows the output of AIC/BIC for different number of components. It shows that we should use 16 number clusters for the EM implementation according to both AIC and BIC criteria.

Metrics	Score
Adjusted Rand Index	0.00306
Adjusted Mutual Information	0.00332
Homogeneity	0.01652
Completeness	0.00375
Silhouette	0.14504

We can see from the table that the clusters are not similar to the class labels. As per the Silhouette value the clusters are not dense.

From the above results we can conclude that EM clustering and K- means clustering have almost similar performance on this dataset. Both algorithms give same number of clusters.

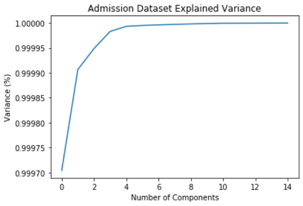
Feature Dimensionality Reduction algorithms

Decision Tree Implementation-

Top Features	Feature Importance
Languages	0.04
Age	0.04
Work Exp	0.04
Volunteer/Leadership	0.04
Certificates/Awards	0.07
TOEFL Score	0.08
SOP	0.09
GRE Score	0.10
LOR	0.11
CGPA	0.13

We can use these top features to build our models for further analysis to get improved results.

PCA Implementation –



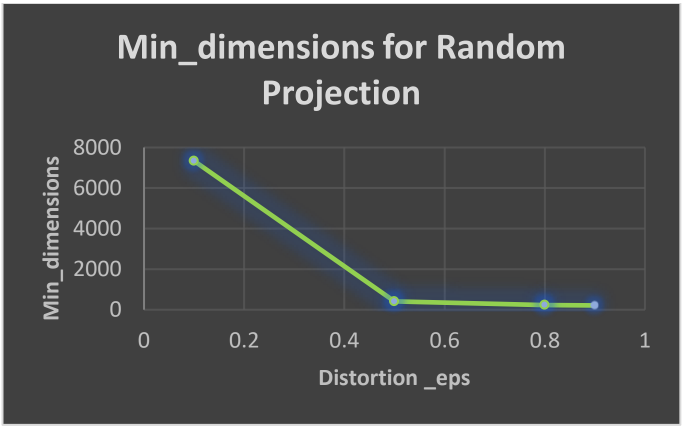
The figure shows that 5 components will give maximum cumulative variance %. So, I will use 5 components for PCA for further analysis on this dataset.

```
original shape:    (5203, 15)
transformed shape: (5203, 5)
```

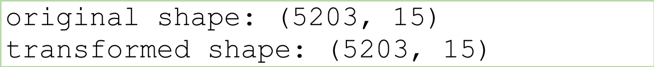
ICA Implementation- Similar implementation as previous dataset

```
original shape: (5203, 15)
transformed shape: (5203, 5)
```

RP implementation- Similar implementation as previous dataset



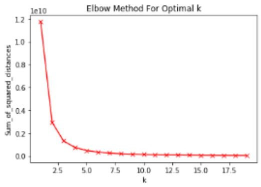
The minimum dimensions according to **Johnson-Lindenstrauss Lemma** for this dataset is 410 for distortion eps = 0.5, As our dimensions is 16. So, I will take 16 as the number of components for RP. Basically, by reducing the number of components RP's performance will decrease and will not help in dimensionality reduction. We can try finding minimum dimensions by reducing the sample size, but this will not be a good option.



```
original shape: (5203, 15)
transformed shape: (5203, 15)
```

(Note: Can use same number of clusters as decided by PCA for easy implementation or can implement Randomized PCA)

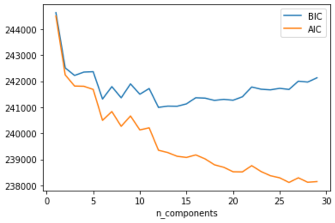
K means implementation after applying dimensionality reduction using PCA –



The figure shows the output of the elbow method. It shows that we should use 2 number clusters for the K means implementation.

Metrics	Score	Interpretations
Homogeneity	0.00751	Decreased after implementing PCA. The score is close to zero, so each cluster does not contain only members of a single class
Silhouette	0.62578	The clusters are dense and well separated as the score is higher

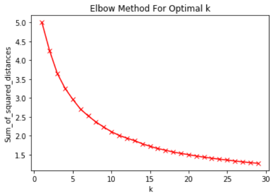
EM implementation after applying dimensionality reduction using PCA –



The figure shows the output of AIC/BIC for different number of components. It shows that we should use around 12 number clusters for the EM implementation according to BIC criteria.

Metrics	Score
Homogeneity	0.04553
Silhouette	0.26031

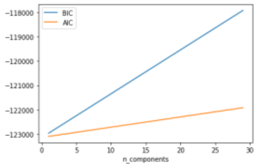
K means implementation after applying dimensionality reduction using ICA –



The figure does not show a clear elbow. I will use 7 number clusters for the K means implementation.

Metrics	Score
Homogeneity	0.01065
Silhouette	0.1761

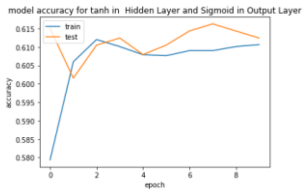
EM implementation after applying dimensionality reduction using ICA –

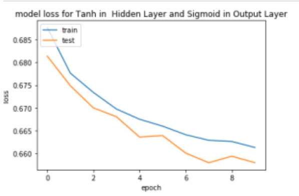


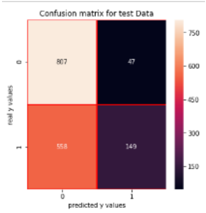
The figure does not explain the valid number of clusters using AIC/BIC criteria after using ICA for dimension reduction.

Neural Network Implementation-

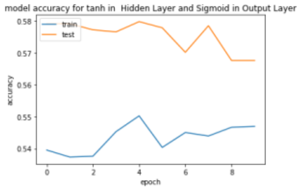
Model without dimension reduction- Used tanh activation function in the hidden and sigmoid in the output layers. loss: 0.6613 - acc : 0.6107 - val_loss: 0.6580 - val_acc: 0.6124

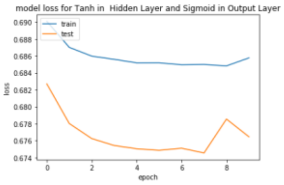


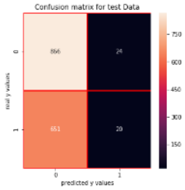




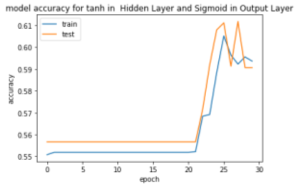
Model results after using PCA for dimensionality reduction– loss: 0.6858 - acc: 0.5470 - val_loss: 0.6765 - val_acc: 0.5676

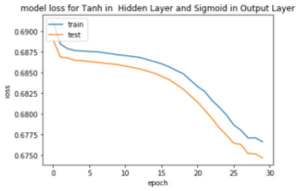


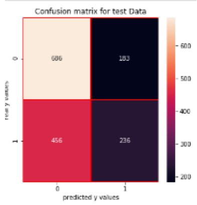




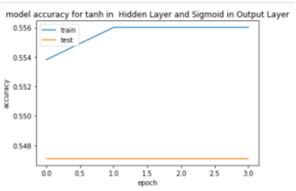
Model results after using ICA for dimensionality reduction– loss: 0.6766 - acc: 0.5936 - val_loss: 0.6747 - val_acc: 0.5906

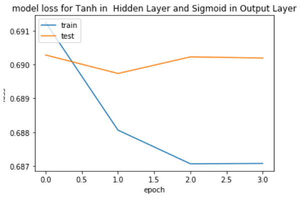


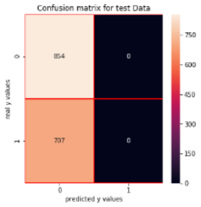




Model results after Using the clustering results from task 1 as the new features and apply neural network learner on this new data consisting of only clustering results as features and class label as the output. –loss: 0.6871 - acc: 0.5560 - val_loss: 0 .6902 - val_acc: 0.5471



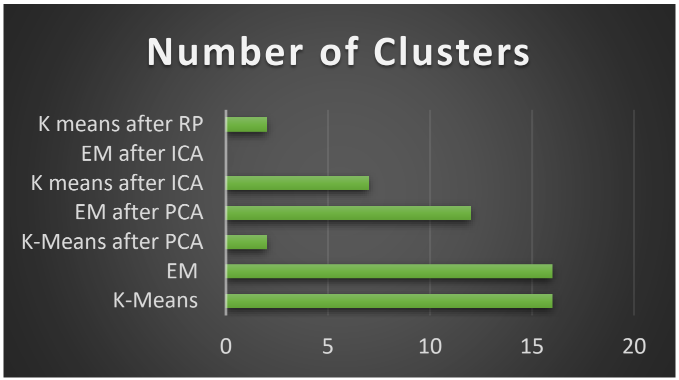


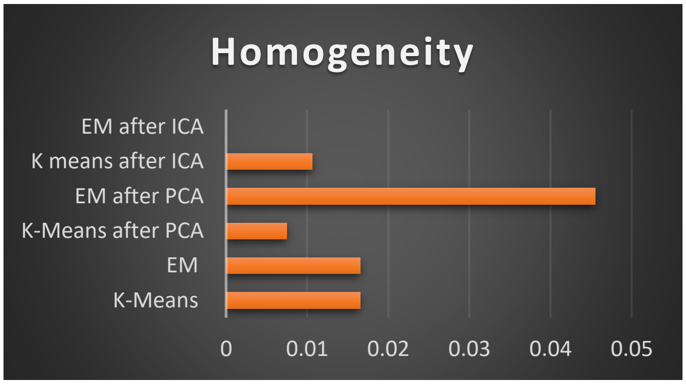


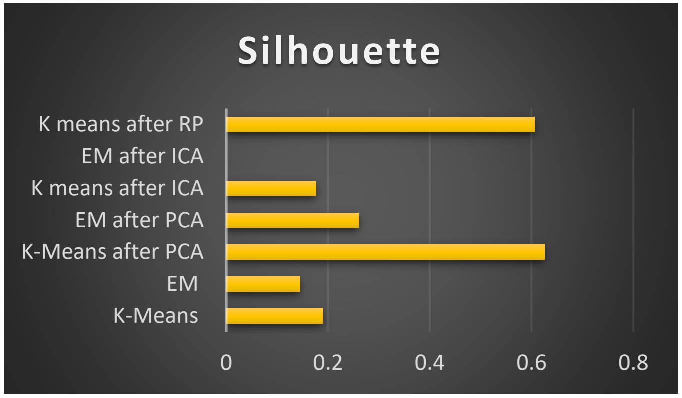
On an average accuracy of the neural network decreased after using different dimensionality reduction methods. Speed does not change much

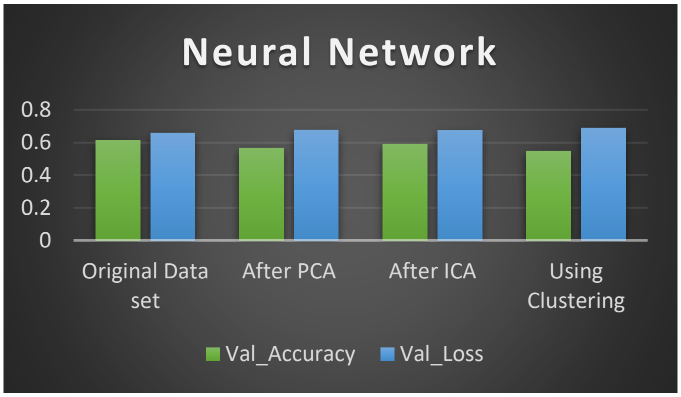
Comparission of Models – The below graphs shows the comparassion of different models that I have experimented with in this assignment for admission dataset.

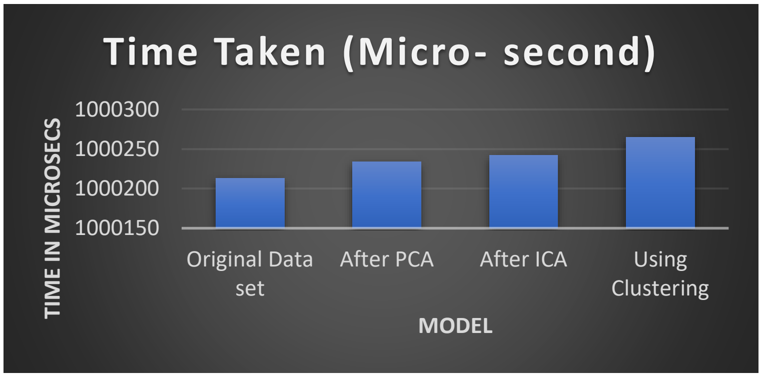
- Got different number of clusters for different experiments that were conducted
- The homogeneity is very less for all the variations of models which shows that points of different nature are also present in the same cluster(very less similarity with the original class lables)
- The Silhouette score ranges from 0.3 to 0.65 , the clusters formed by K means after using PCA and RP are comparitively denser
- After applying dimensionality reduction on this data set the accracy of the neural network model decreased for all the variations
- We can also compare the shapes of the final dataset obtained depending on the number of clusters to have a better visualization of the data











Conclusion-

There are several drawbacks of different clustering algorithms dude to which it does not produce good results for all types of datasets:

- Doesn't allow development of an optimal set of clusters and for effective results, you should decide on the clusters before.
- Clustering gives varying results on different runs of an algorithm. A random choice of cluster patterns yields different clustering results resulting in inconsistency.
- It produces cluster with uniform size even when the input data has different sizes.
- The way in which data is ordered in building the algorithm affects the results of the data set.
- Changing or rescaling the dataset either through normalization or standardization will completely change the results.
- When dealing with a large dataset, conducting a dendrogram technique will crash the computer due to a lot of computational load and Ram limits.
- Algorithm can be performed in numerical data only.
- K-means clustering technique assumes that we deal with spherical clusters and each cluster has equal numbers for observations. The spherical assumptions must be satisfied. The algorithm can't work with clusters of unusual size.
- It is difficult to predict the k-values or the number of clusters. It is also difficult to compare the quality of the produced clusters.