Prepared by: KHUSHBOO BHATIA (kxb190007) Date: 05/01/2020

BUAN 6341 Applied Machine Learning

ASSIGNMENT NO 4

Introduction:

In this report, I have implemented **K-means** and **Expectation Maximization** and in addition implemented the feature dimensionality reduction algorithms like PCA, ICA and Randomized Projections. I have used **Rain in Australia** dataset as my second dataset for this project. This Data set is obtained from Kaggle, link to the dataset is given below –

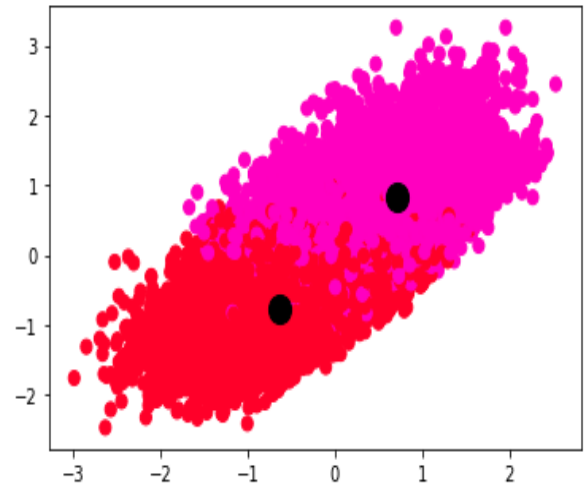
https://www.kaggle.com/jsphyg/weather-dataset-rattle-package

Rain in Australia Data Set

Task 1: Running the Clustering Algorithms

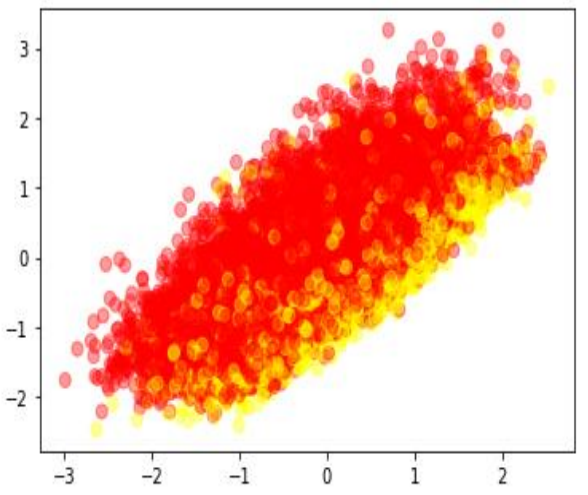
K-Means and Expectation Maximization clustering algorithms are used without any dimensionality reduction. Features used are Maxtemp, Mintemp, Rainfall, WindGustSpeed, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Temp9am, Temp3pm, Today_Rain.

```
[[4627 4742]
 [1593 1038]]
0.47208333333333335
```



(K- Means)

```
[[4325 5044]
 [1123 1508]]
0.4860833333333333

<matplotlib.collections.PathCollection at 0x1719
```



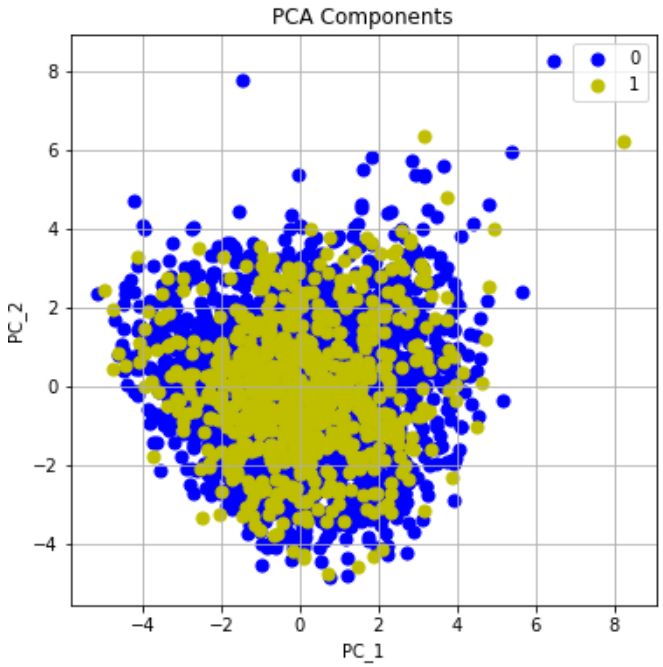
(Expectation Maximization)

The Accuracy for K-Means is 47.2% and Expectation Maximization is 48.6%. Poor accuracy score shows that this data is not suitable for K-Means and Expectation Maximization and therefore clusters are not well separated.

Task 2: Applying the Dimensionality Reduction algorithms

Forward feature selection algorithm is used for dimensionality reduction. Filtered features are Maxtemp, Mintemp, WindGustSpeed, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Today_Rain.

Principal Component Analysis(PCA) is used on filtered features after Normalization. Other feature transformation algorithms such as Independent Component Analysis (ICA) and Random Component Analysis(RCA). Different feature sets are obtained for PCA, ICA and RCA. Below plot shows the PC1 and PC2 components in PCA, which explains the maximum variances in our dataset.

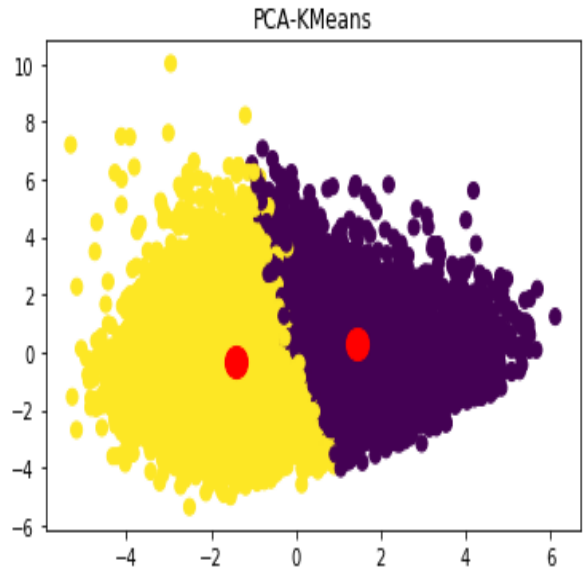


This graph shows that our data is not linearly separable and therefore separate clusters are not formed.

Task 3: Running the Clustering Algorithms after applying Dimensionality Reduction

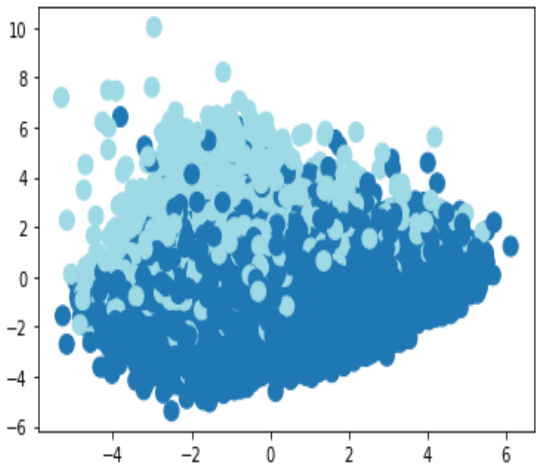
Principal Component Analysis(PCA): PCA in conjunction with K-Means is a powerful method for visualizing high dimensional data. PCA helps to reduce the no of features while preserving the variance. Expectation Maximization assigns a probability distribution to each instance.

```
0.4495
```



(K- Means)

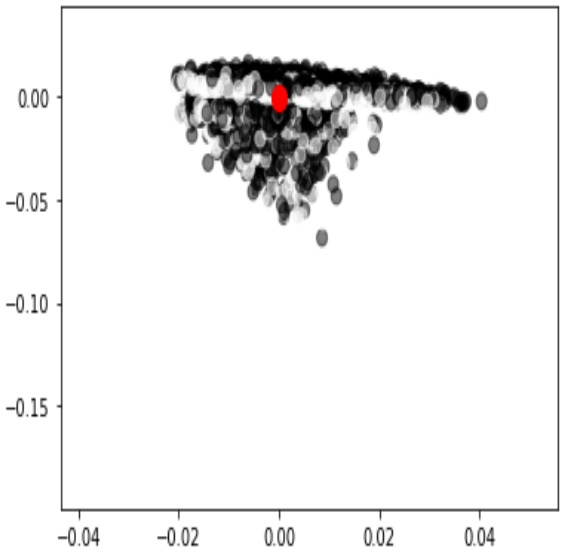
```
[[5384 3985]
 [ 398 2233]]
0.63475

<matplotlib.collections.PathCollection at 0x1718
```



(Expectation Maximization)

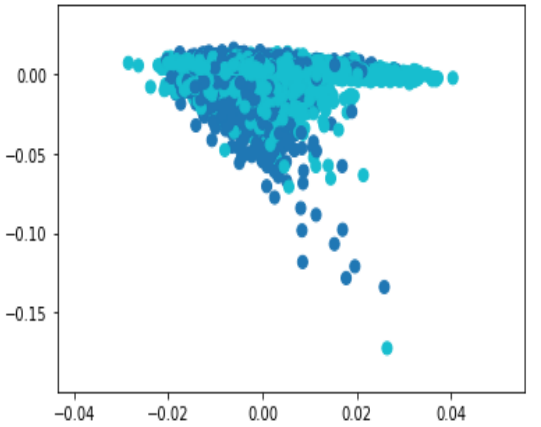
<u>Independent Component Analysis(ICA):</u> ICA attempts to decompose a multivariate signal into independent Non-Gaussian signals.

```
0.3884166666666667
```



(K- Means)

```
[[7817 1552]
 [1618 1013]]
0.7358333333333333

<matplotlib.collections.PathCollection at 0x17197bc
```

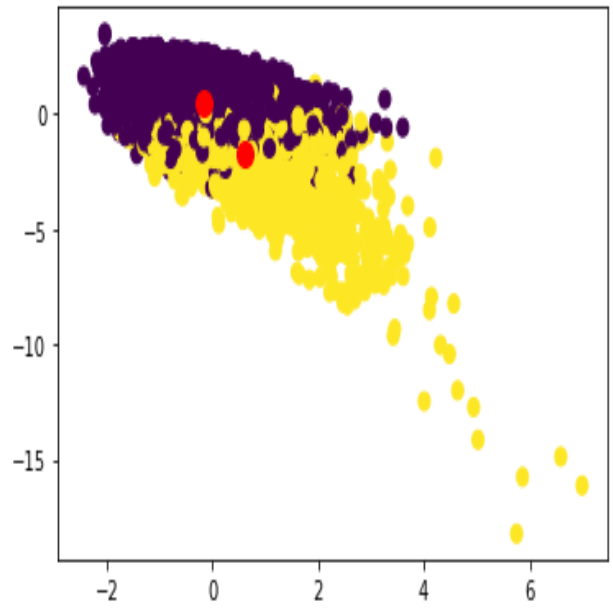


(Expectation Maximization)

The Accuracy is obtained 38.84% for K-Means and 73.58% for EM after ICA.

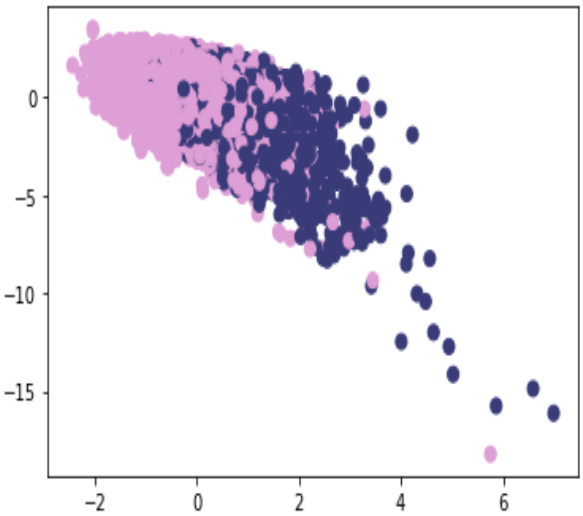
Randomized Projection: Randomized Projection or RCA picks random directions and projects the data on to these random directions.

```
0.7803333333333333
```



(K- Means)

```
[[7960 1409]
 [1419 1212]]
0.7643333333333333

<matplotlib.collections.PathCollection at 0x17197
```



(Expectation Maximization)

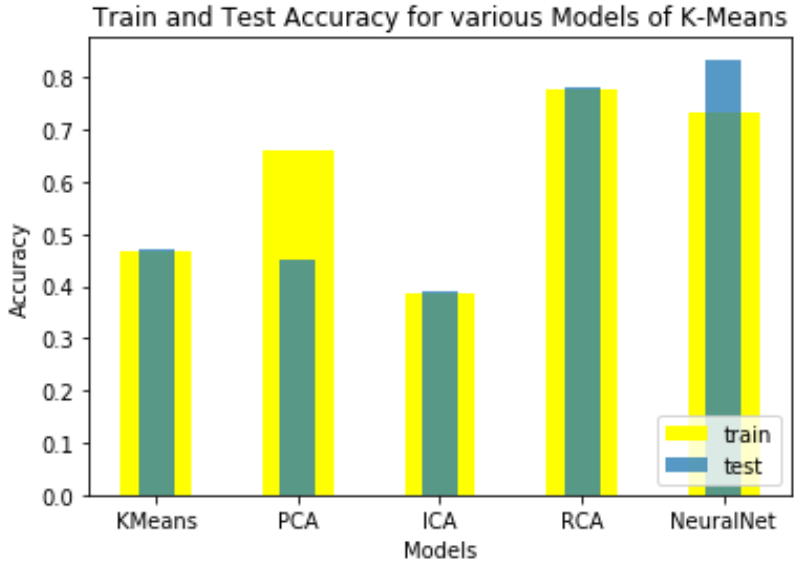
PCA provides the compact cluster whereas ICA provides dispersed cluster. And in case of RCA, it provides optimum clusters of compact and less dispersed cluster. The Accuracy is obtained 78.03% for K-Means and 76.43% for EM after Randomized Projection.

For this Dataset, PCA performed best out of all feature transformation algorithm. It retains the most information from the feature set in most of the cases.

Task 4: Running Neural Network after Dimensionality Reduction

Dimensionality Reduction plays a really important role in machine learning, especially when we have large input space. Artificial Neural Network is implemented on transformed features after dimensionality reduction. Below graph represents the Train and Test Accuracy for all the algorithms using K-Means and Expectation Maximization. The Test accuracy for Neural Network is 83.48%. The confusion matrix shows really good results and lead to commit less misclassification.

```
Text(0.5, 1.0, 'Train and Test Accuracy for various Models of K-Means')
```



From graph, we can see that Neural Network performed much better as comparison with other algorithms. Its best performance is due to the fact that Neural Network can form complex decision boundaries as hypothesis, Also Neural Network runs faster. ICA performed worst as compare to other algorithms.

Task 5: Running Neural Network with features from K-Means & Expectation Maximization

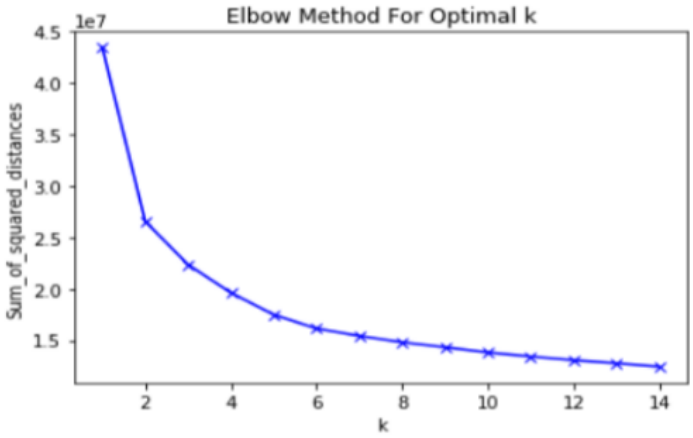
We performed Neural Network analysis with features from K-Means & Expectation Maximization. The Train and Test accuracy of model is 76.97% and 78.27%.

```
              precision    recall  f1-score   support

         0.0       0.79      0.96      0.87     31214
         1.0       0.40      0.10      0.16      8786

    accuracy                           0.77     40000
   macro avg       0.60      0.53      0.51     40000
weighted avg       0.71      0.77      0.71     40000

0.769775
              precision    recall  f1-score   support

         0.0       0.80      0.96      0.87     21845
         1.0       0.52      0.14      0.22      6155

    accuracy                           0.78     28000
   macro avg       0.66      0.55      0.55     28000
weighted avg       0.74      0.78      0.73     28000

0.78275
```

In clustering algorithms, K-means with PCA performs the best on the dataset. Overall Neural Networks with PCA features performs exceptionally well giving a significant boost to the accuracy and enabling us to form complicated hypothesis functions.

Optimum Value of K

The optimum value of K is chosen using the Elbow method. As the value of K increases, there will be fewer elements in the cluster.

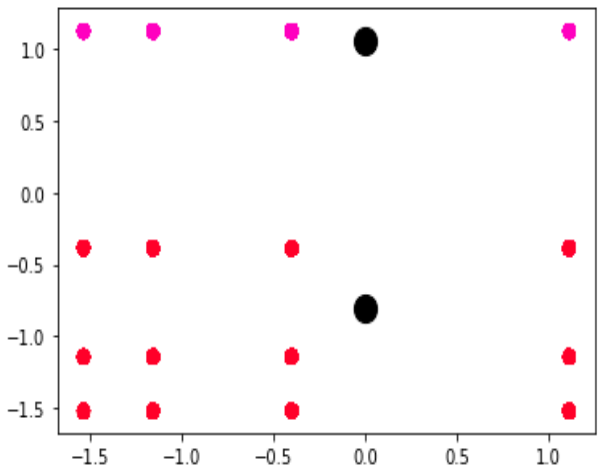


SGEMM GPU kernel performance Data Set

Task 1: Running the Clustering Algorithms

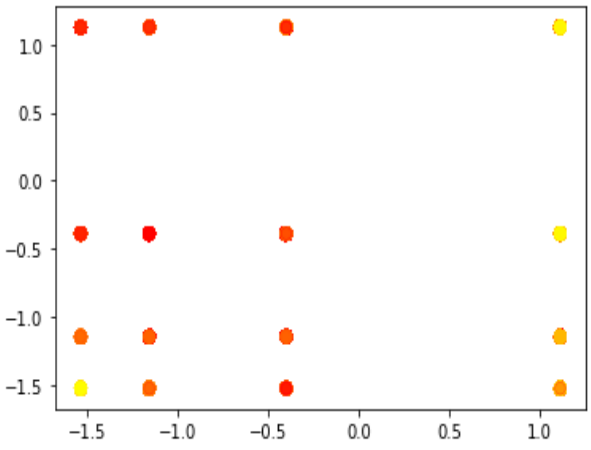
K-Means and Expectation Maximization clustering algorithms are used without any dimensionality reduction. Features used are MWG, NWG, KWG, MDIMC, NDIMC, MDIMA, NDIMB, KWI, VWM, VWN, STRM, STRN, SA, SB.

```
[[4121 1815]
 [2746 3318]]
0.6199166666666667
```



(K- Means)

```
[[ 700 5236]
 [1579 4485]]
0.4320833333333333

<matplotlib.collections.PathCollection at 0x1dcd3c
```



(Expectation Maximization)

The accuracy is 61.9% and 43.2% for K- Means and Expectation Maximization respectively. K- Means algorithm performs the better of both the models.

Task 2: Applying the Dimensionality Reduction algorithms

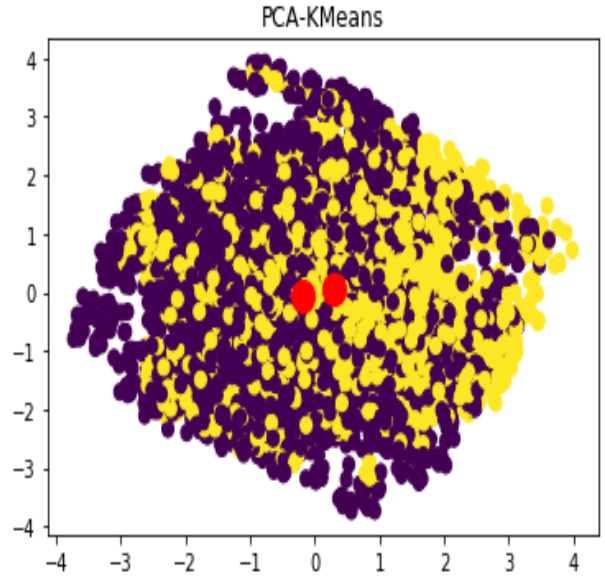
Forward feature selection algorithm is used for dimensionality reduction. Filtered features are MWG, NWG, KWG, MDIMC, NDIMC, MDIMA, KWI, VWM, VWN, STRM, STRN, SA, SB.

Principal Component Analysis(PCA) is used on filtered features after Normalization. Other feature transformation algorithms such as Independent Component Analysis (ICA) and Random Component Analysis(RCA). Different feature sets are obtained for PCA, ICA and RCA.

Task 3: Running the Clustering Algorithms after applying Dimensionality Reduction

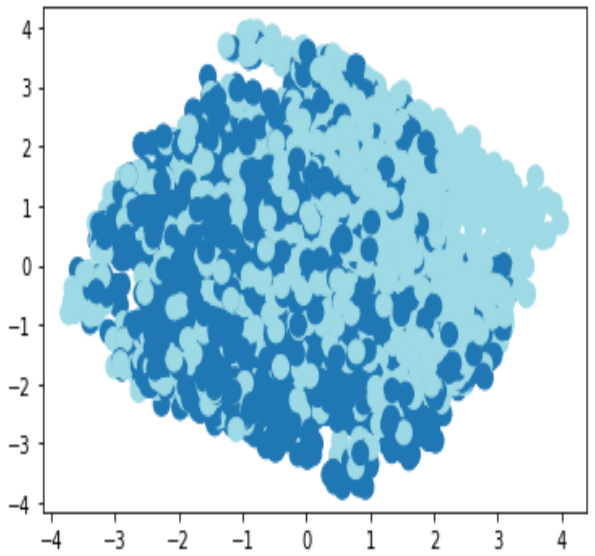
Principal Component Analysis(PCA):

```
0.4800833333333333
```



(K- Means)

```
[[ 543 5393]
 [1867 4197]]
0.395

<matplotlib.collections.PathCollection at 0x1dcd
```



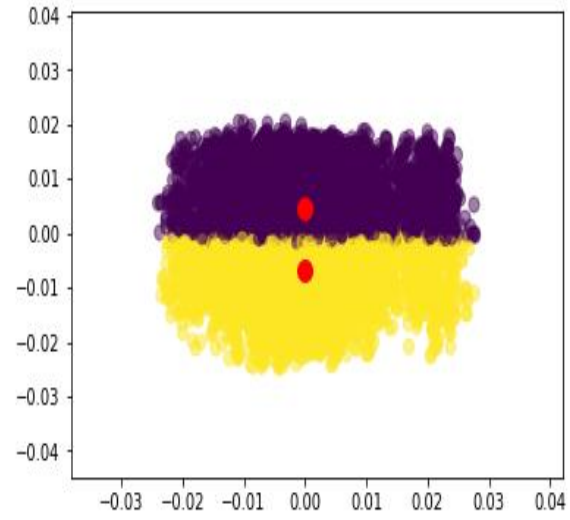
(Expectation Maximization)

The Accuracy is obtained 48% for K-Means and 39.5% for EM after PCA.

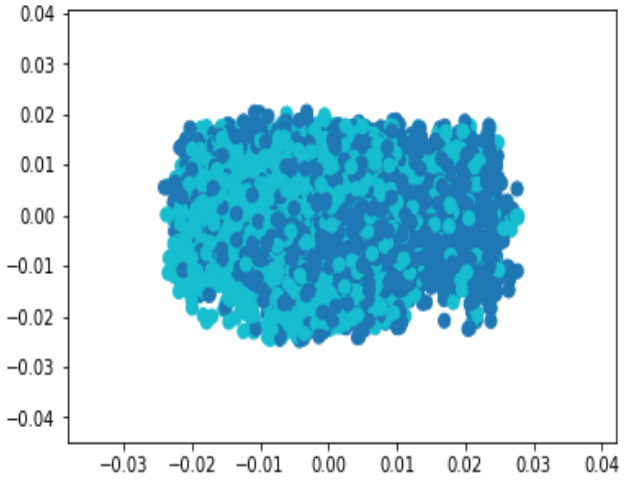
Neither of the algorithm K – Means and EM does not have ability to detect the outliers from the detected clusters. The K – Means algorithm attempts to detect clusters within the dataset under the optimization criteria that the sum of the inter-cluster variances is minimized.

Independent Component Analysis(ICA):

```
0.43416666666666665
```



```
[[4387 1549]
 [4372 1692]]
0.5065833333333334
<matplotlib.collections.PathCollection at 0x1dcd406
```

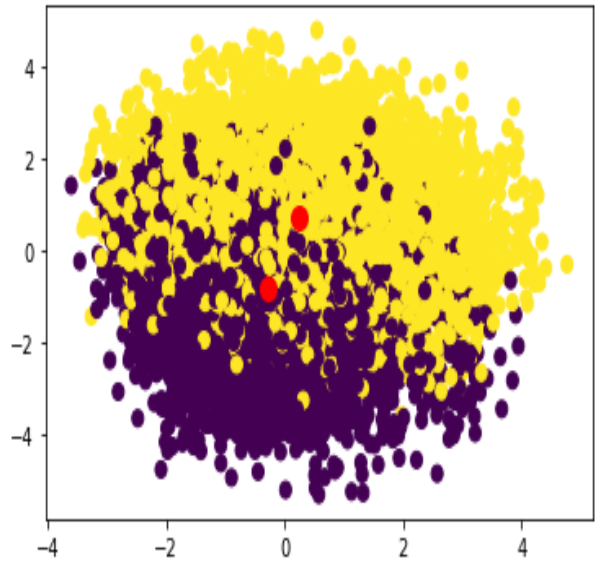


(K- Means) **(Expectation Maximization)**

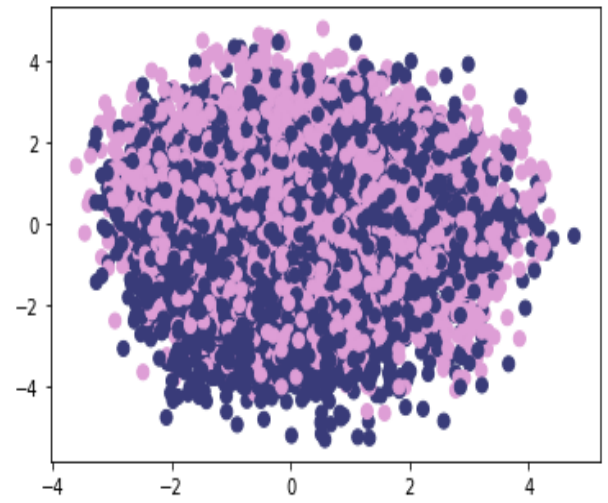
The Accuracy is obtained 43.41% for K-Means and 50.65% for EM after ICA.

Randomized Projection: The Accuracy is obtained 47.25% for K-Means and 51.07% for EM after PCA.

```
0.4725
```



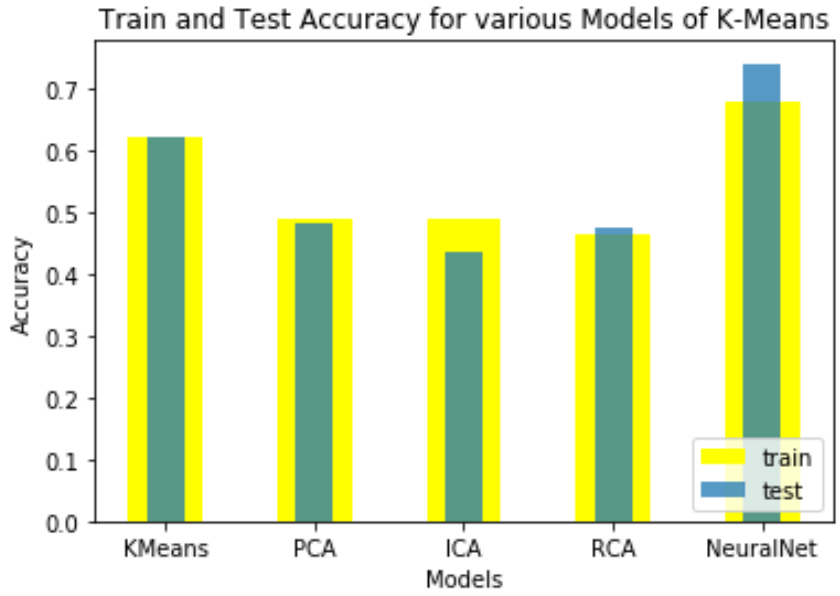
```
[[1625 4311]
 [1560 4504]]
0.51075
<matplotlib.collections.PathCollection at 0x1dcd
```

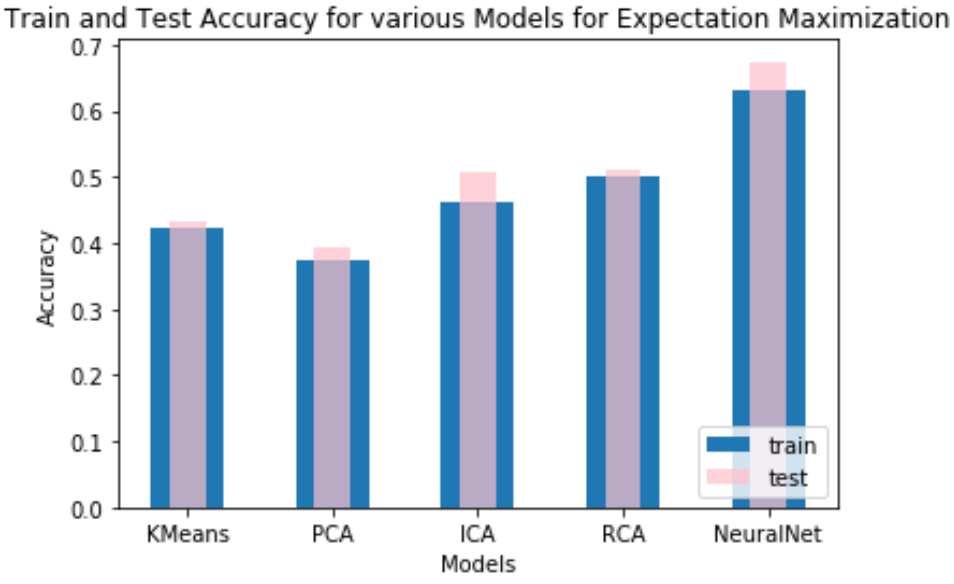


(K- Means) **(Expectation Maximization)**

Task 4: Running Neural Network after Dimensionality Reduction

Artificial Neural Network is implemented on transformed features after dimensionality reduction. Below graph represents the Train and Test Accuracy for all the algorithms using K-Means and Expectation Maximization. The Test accuracy for Neural Network is 67.44%.





From graph, we can see that Neural Network performed much better as comparison with other algorithms. Its best performance is due to the fact that Neural Network can form complex decision boundaries as hypothesis, Also Neural Network runs faster. ICA performed worst as compare to other algorithms.

Task 5: Running Neural Network with features from K-Means & Expectation Maximization

We performed Neural Network analysis with features from K-Means & Expectation Maximization. The Train and Test accuracy of model is 63.26 % and 67.44 %.

```
              precision    recall  f1-score   support

         0.0       0.65      0.56      0.60     19880
         1.0       0.62      0.71      0.66     20120

    accuracy                           0.63     40000
   macro avg       0.64      0.63      0.63     40000
weighted avg       0.64      0.63      0.63     40000

0.632675
              precision    recall  f1-score   support

         0.0       0.66      0.73      0.69     13944
         1.0       0.70      0.62      0.66     14056

    accuracy                           0.67     28000
   macro avg       0.68      0.67      0.67     28000
weighted avg       0.68      0.67      0.67     28000

0.6744642857142857
```

Neural Networks with PCA features perform exceptionally well giving a significant boost to the accuracy and enabling us to form complicated hypothesis functions.