**Assignment 4**

**Introduction:**

The objective of this project is to analyze the clusters in our datasets for high intensive GPU loads and telecom customer churn prediction. We will be using k means and expectation maximization algorithms to split out datasets into 2 clusters to study our two binary classes (0 and 1). Plotting the clusters will help us visualize and study the performance of these algorithms.

**Part 1**

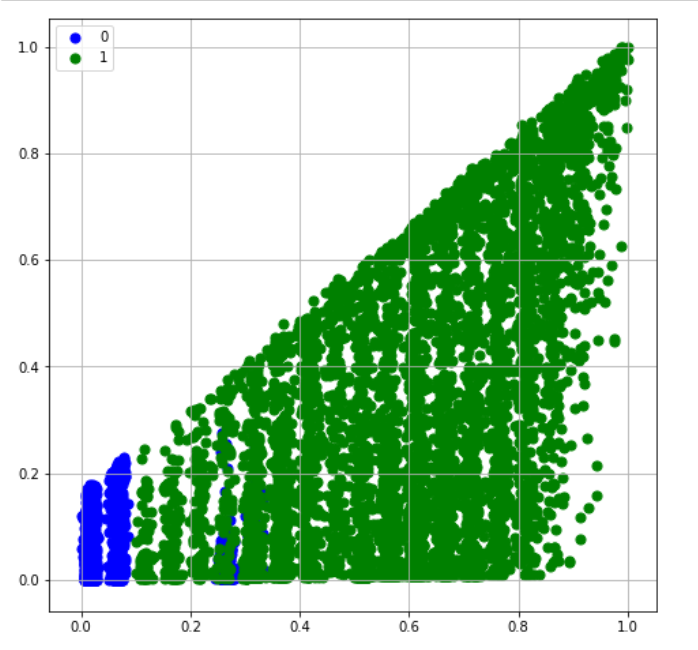
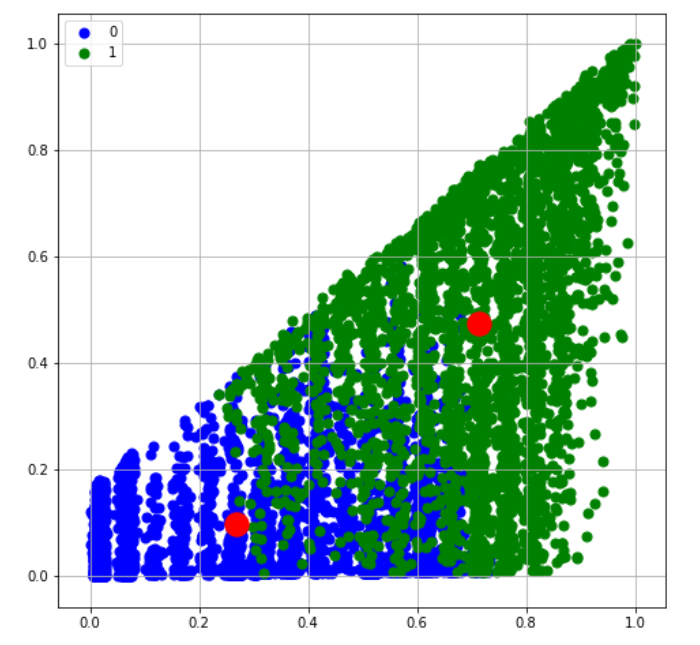
**Telecom Customer Churn Prediction**

**Dataset:**

The dataset we will be using for this project has a total of 21 features and 7043 records. We will be dropping customerId and PaymentMethod from the dataset and will be using churn as our target variable. Partner, Dependents, PhoneService, PaperlessBilling, Churn, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies variables are converted to binary. Dummy variables have been added to convert InternetService, Contract and Gender to categorical variables. After the above changes we have 20 features we will be using as our independent variables to predict out target variable which is churn.

**Experiment 1:**

In this experiment we will be running both k means and expectation maximization with fixed number of clusters as 2. We will be running with 2 clusters since our dataset is being used for binary classification. So, by running the k means and EM clusters and plotting them against any two features we will be able to observe the spread of our two binary classes.



Since we already know that the 2 classes are equally distributed in our dataset, we can see that the above clusters do not match our expected distribution. We can check this inference by comparing the class labels from the clustering algorithm with our existing target variable, to see their accuracy.

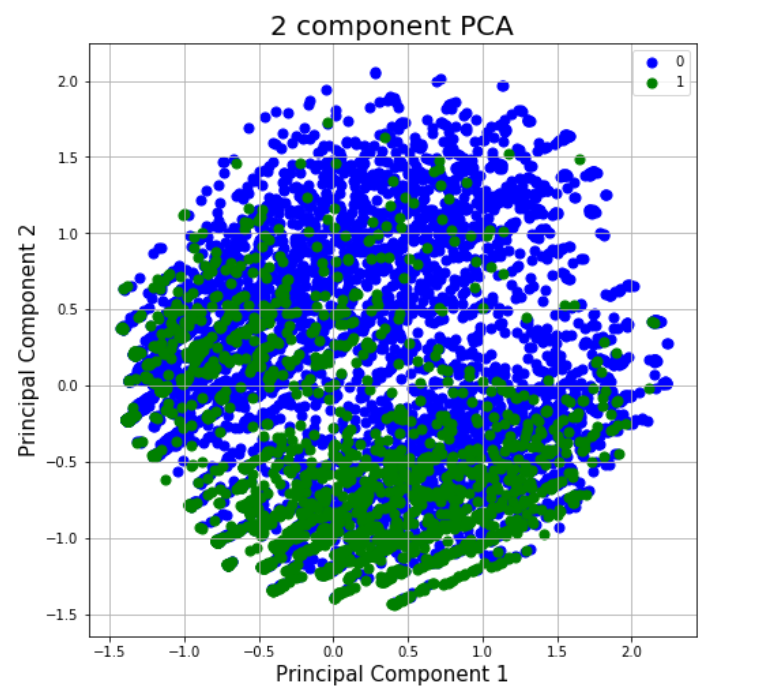
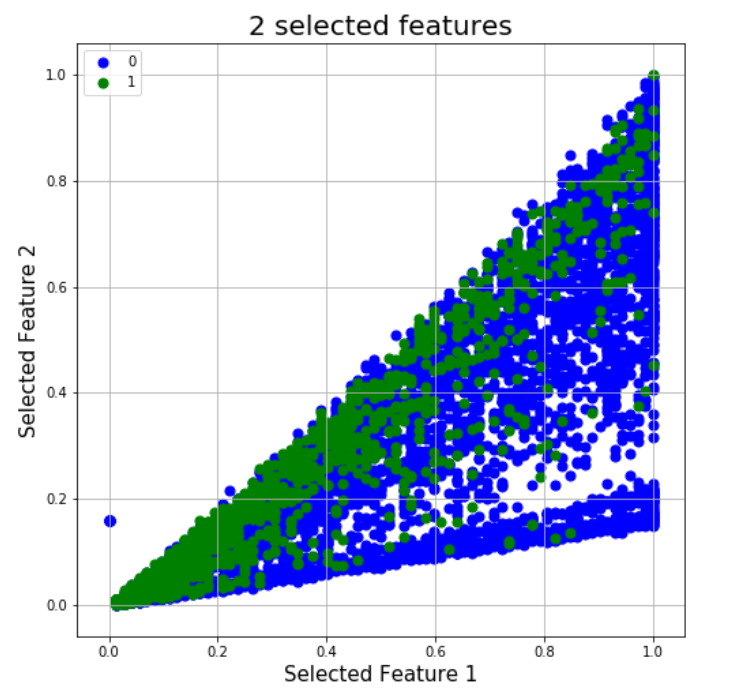
|  |  |
| --- | --- |
| Algorithm | Accuracy |
| K means | 52.12% |
| Expectation maximization | 45.70% |

From the above table we can see that K means and EM can only explain half of the variations in our dataset.

**Experiment 2:**

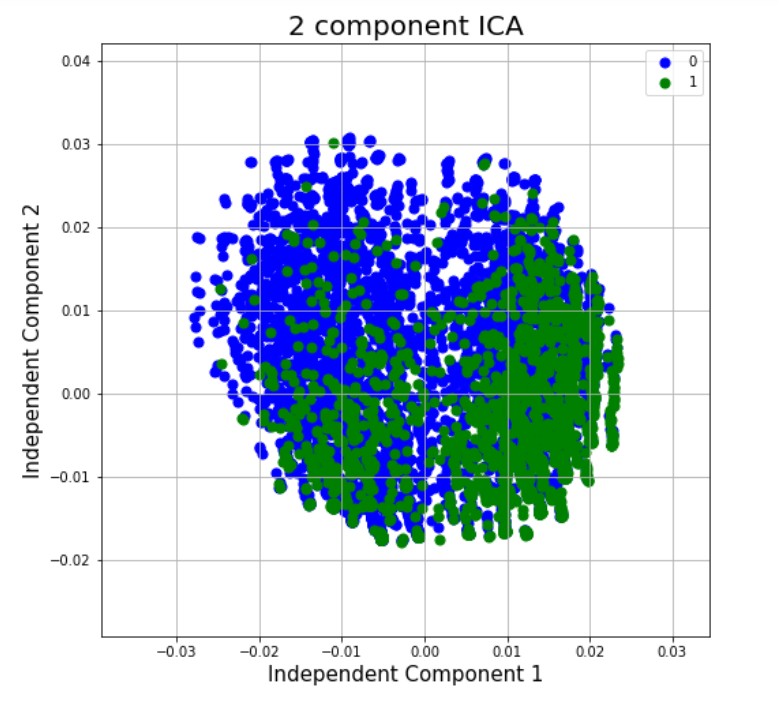
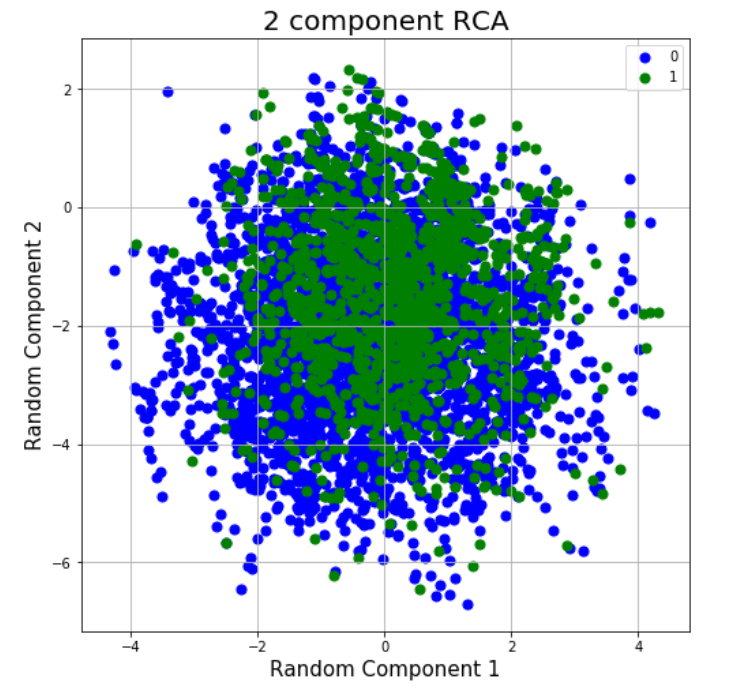
In this experiment we will be running various dimensionality reduction on our dataset using feature selection and feature transformation. Will be plotting our observations in order to understand their impact on our dataset.

We will first start with feature selection using a decision tree and PCA:



From the above graphs we can see that the distributions of our classes are not linearly separable. Although the density of the distributions varies across the plane, in general the classes are evenly spread across the terrain.

Now plotting distributions of ICA and RCA:

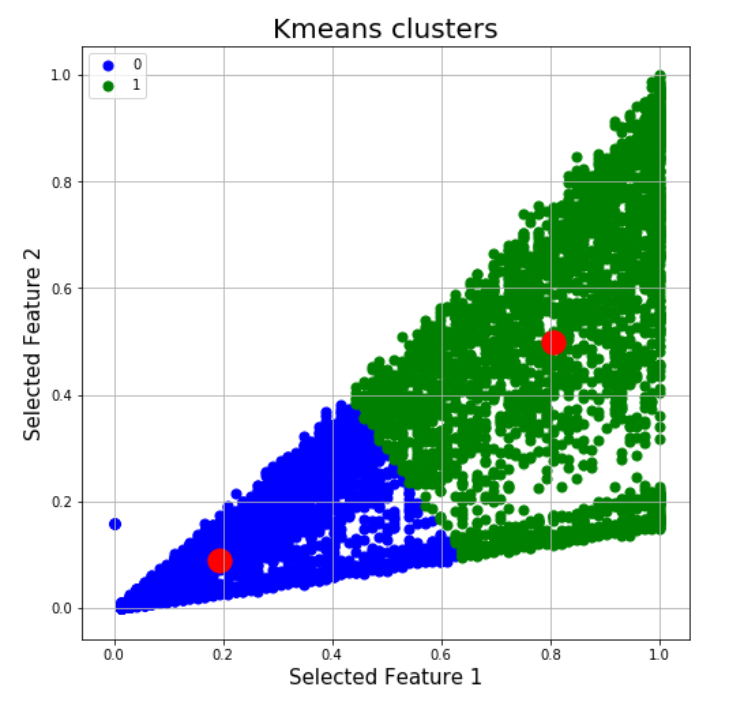
 

Similar to the feature selection and PCA, the data points seem to be evenly spread and not linearly separable. This might suggest that the performance of our clustering algorithms might not be good at classifying this dataset. The density of ICA data seems to be more compact in comparison to PCA, PCA also seems to be a little sparse in density when compared to RCA.

**Experiment 3:**

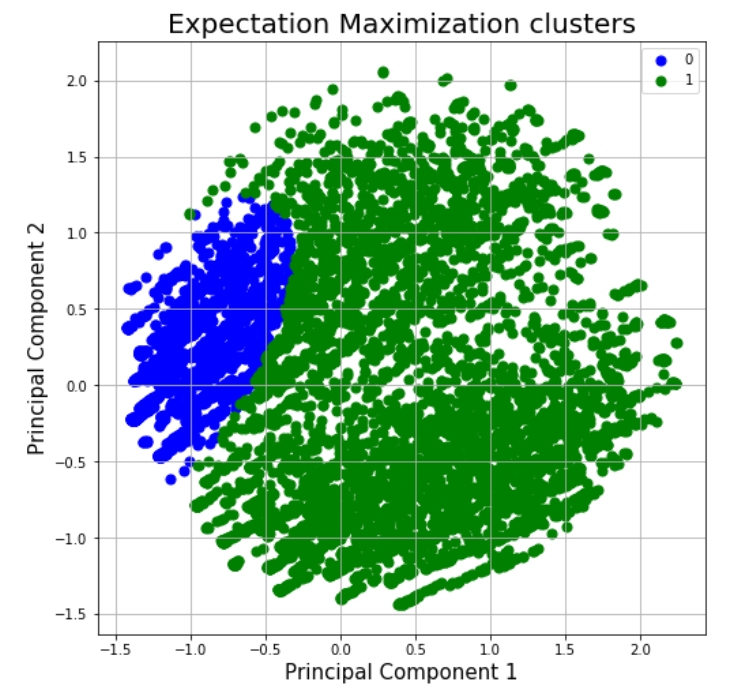
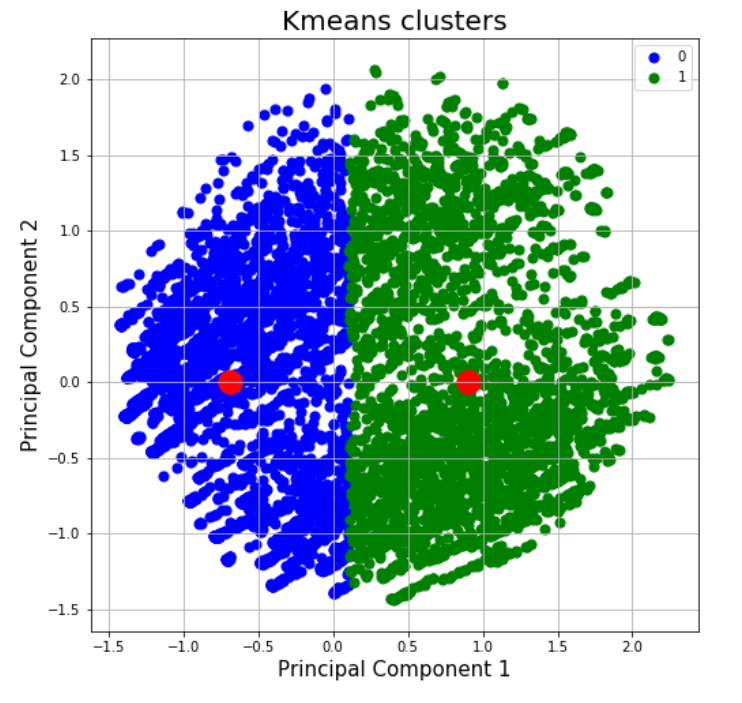
In this experiment we will be running k means and EM algorithms on the dimensionally reduced dataset from the previous experiment. We will be plotting the resulting clusters to analyze the performance of the algorithms. We will also be using accuracy as a metric to compare their relative performances.

Feature Selection:



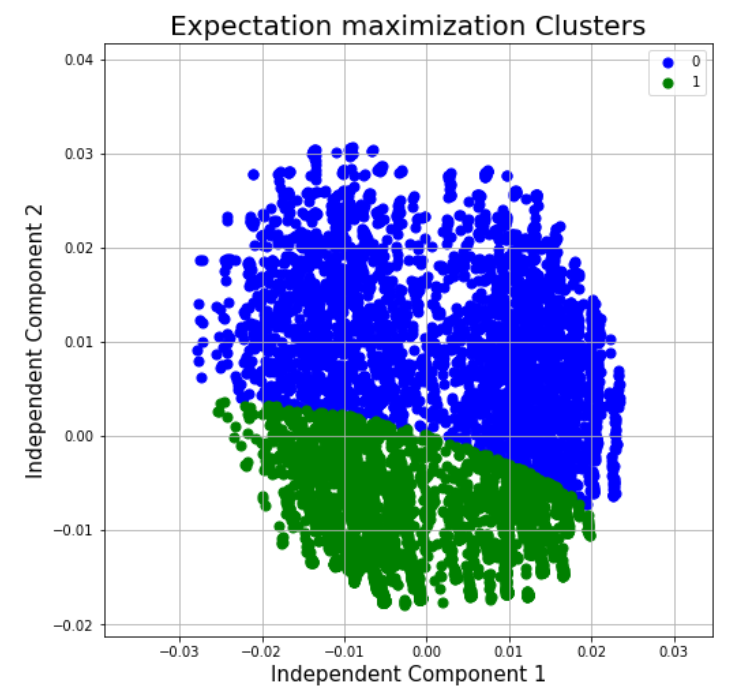
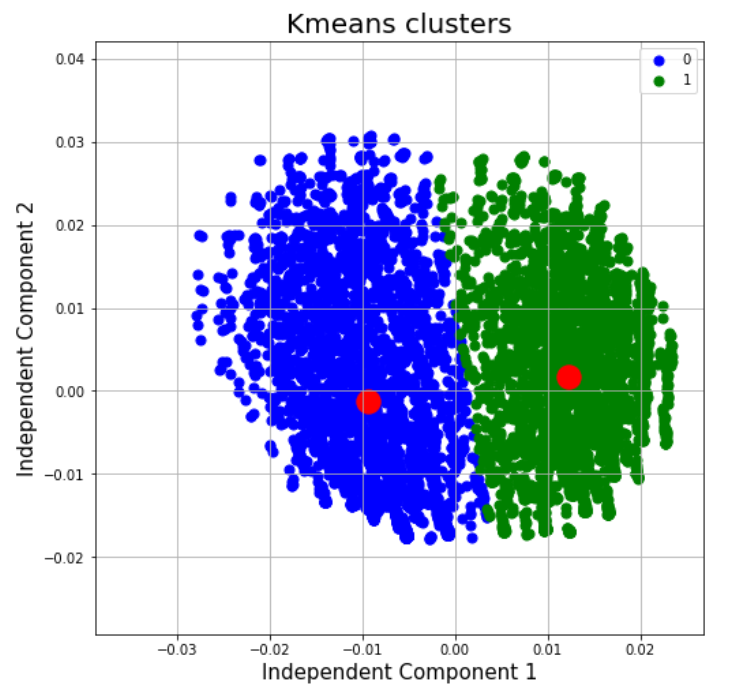
Comparing the above plots to the plot for feature selection from experiment 2, we can conclude that k means and EM do not do a good job of isolating the 2 classes. Since the accuracy of the k means model is 42.41% and EM model is 28.26% this confirms our inference.

PCA:



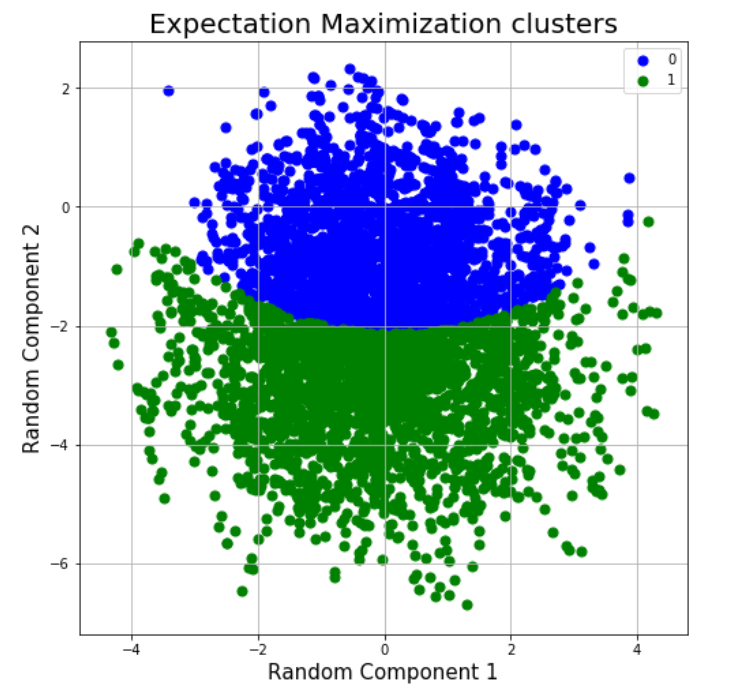
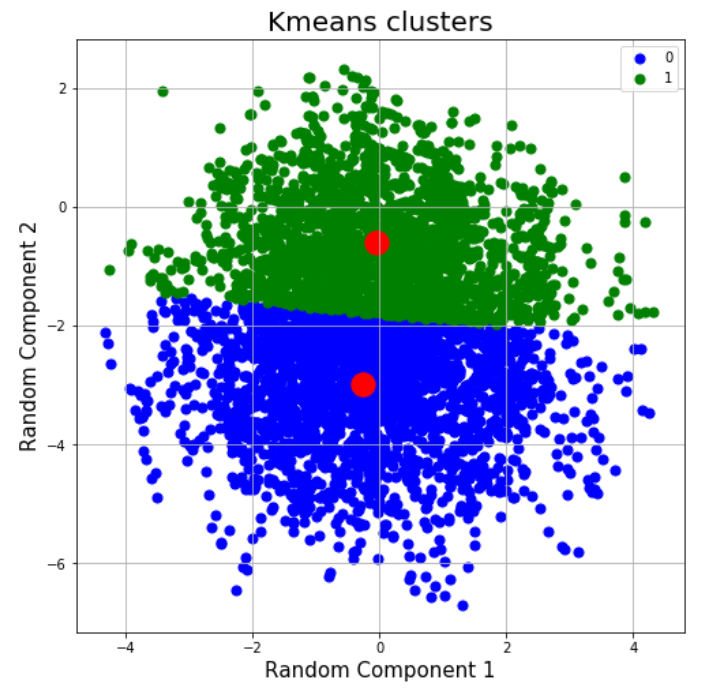
Similar to the feature selection comparing this graph to plot from experiment 2 for PCA, we can see these models only explain around half of the variations between the 2 classes. Since the accuracy for these models are 52.18% and 47.58% respectively. This confirms our assertion.

ICA:



Similar to PCA these plots so that k means and EM can explain around 50% of the data. The model accuracy metric which are 66.65% and 51.91% respectively, also confirms this inference.

RCA:

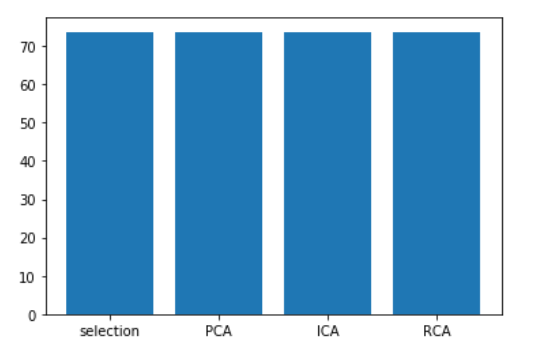


The performance of the above models also seems to explain only half of the model. This is consistence with the model accuracy metric of 51.91% and 49.41% of the models.

From the above metric we can see that k means performs better than EM for this dataset.

**Experiment 4:**

For this experiment we will using the same dimensionality reduction techniques from the above experiments and we will be observing their impact on a neural network. The neural network we will be using for this study will have one hidden layer with 128 nodes and one output layer with 2 nodes, all running sigmoid activation functions. Running a model using this neural network on the dimensionally reduced dataset from the previous experiments we get the below results:



From the above graph we can see that the accuracy of the neural network is 73.71% irrespective of the which dimensionality reduction tool is used.

**Experiment 5:**

From this experiment we will running the same neural network from the previous experiment on the outputs from our k means and EM models from experiment 1. Running this model, we get an average accuracy 73.71% which is the same as the models from experiment 4. So, we can conclude that the k means and EM cannot be used to improve our results, since they do not add additional meaning to the data.

**Part 2**

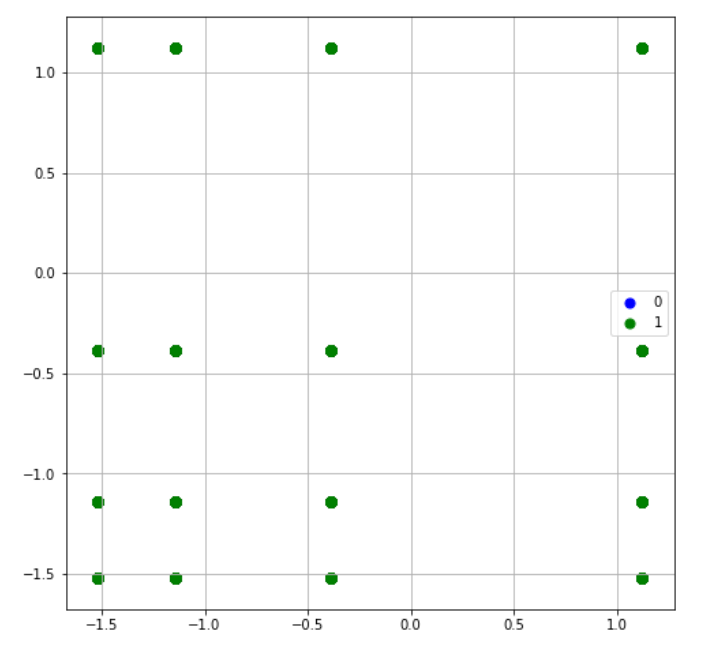
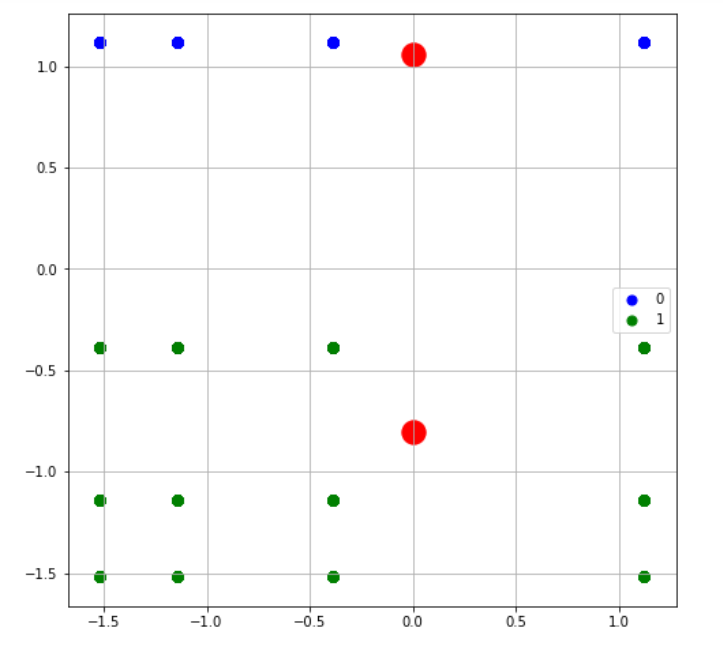
**GPU Average Runtime Prediction**

**Dataset:**

The dataset we will be using for this project has a total of 18 features and 241600 records. We will be taking the average of 4 of these features Run1, Run2, Run3 and Run4 as AverageRun and will be using the remaining 14 for our prediction. In order to convert the average GPU, run time into binary classes, we create a new variable to indicate if the average run is greater than the median. This splits our data almost equally with 50% of ones and 50% zeroes.

**Experiment 1:**

In this experiment we will be running both k means and expectation maximization to with fixed number of clusters as 2. We will be running with 2 clusters since our dataset is being used for binary classification. So, by running the k means and EM clusters and plotting them against any two features we will be able to observe the spread of our two binary classes.

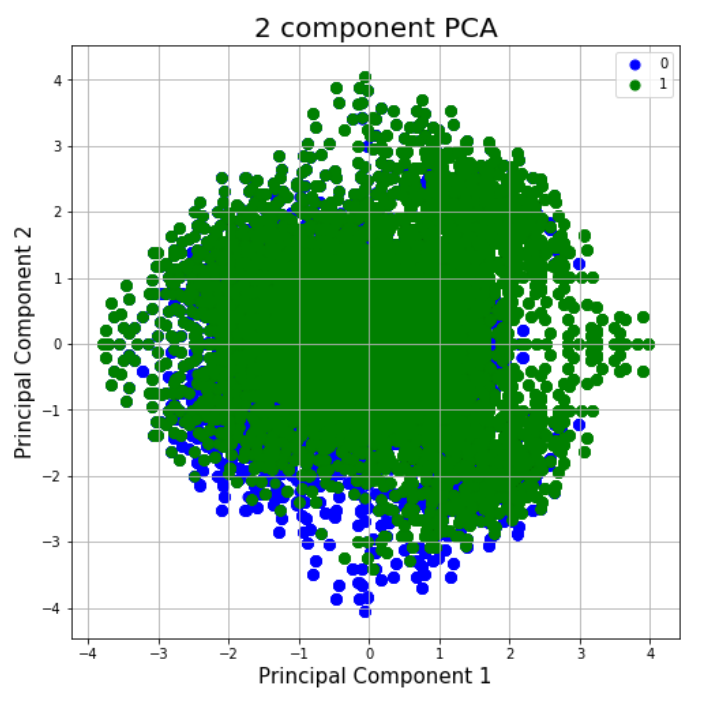
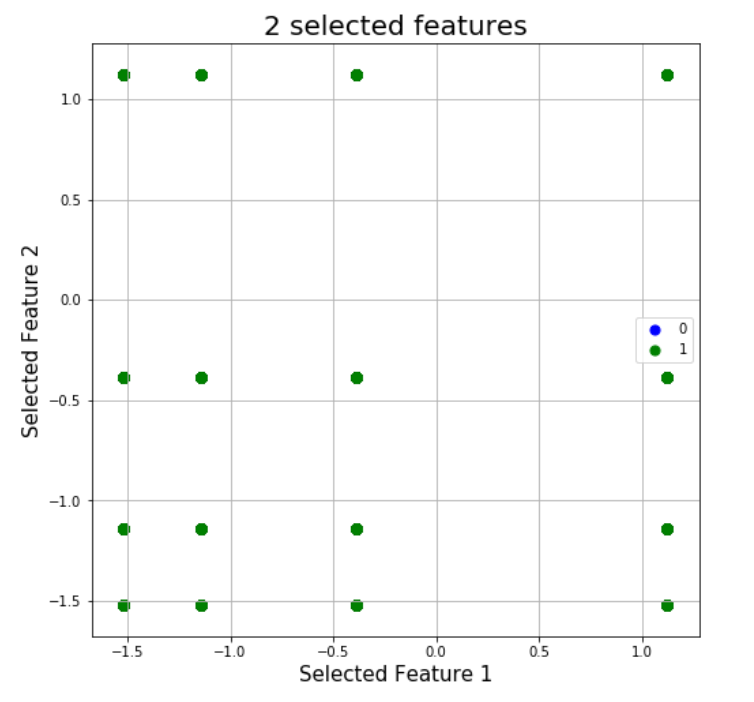


From the above plots we can see that the clusters are not of equal size as we expect, since the dataset is split equally on the binary classes. Therefore, we can infer that these models will not perform well on this dataset, as they do not explain the data well. This is backed by our model accuracies with are 37.49% and 48.94% respectively.

**Experiment 2:**

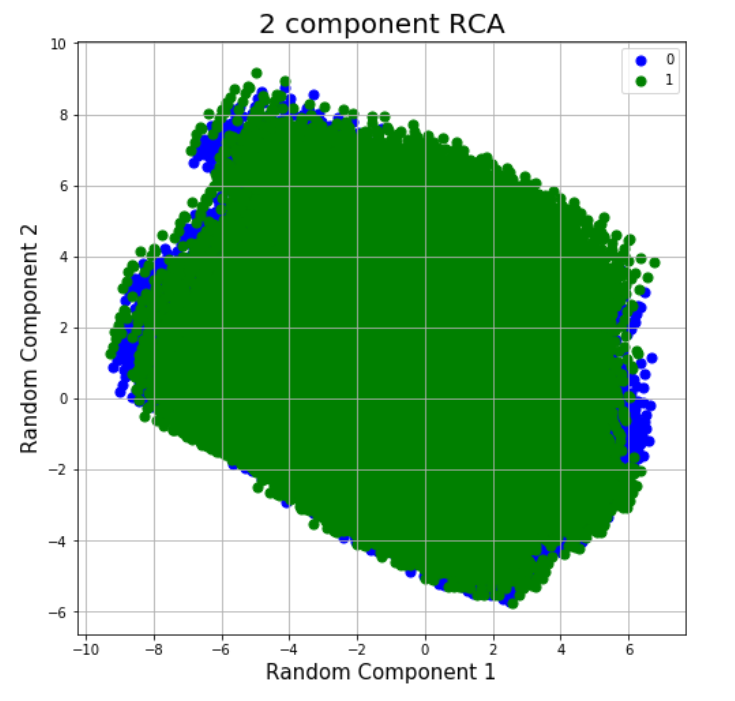
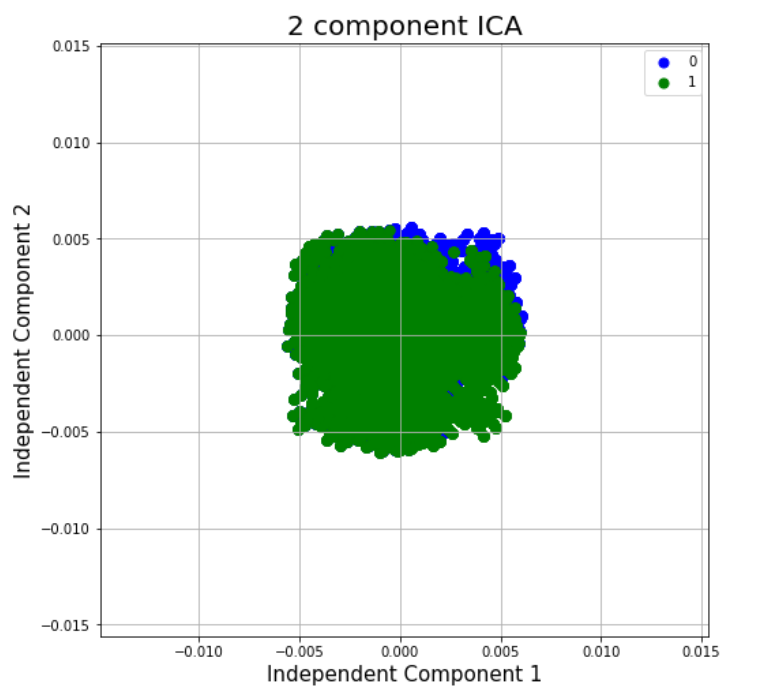
In this experiment we will be running various dimensionality reduction on our dataset using feature selection and feature transformation. Will be plotting our observations to understand their impact on our dataset.

We will first start with feature selection using a decision tree and PCA:



From the above plots we can see that the data points for both the classes are evenly spread in the case of PCA. As for the feature selection plot, looks like the selected features do not have sufficient information regarding 0 classes.

Now plotting distributions of ICA and RCA:



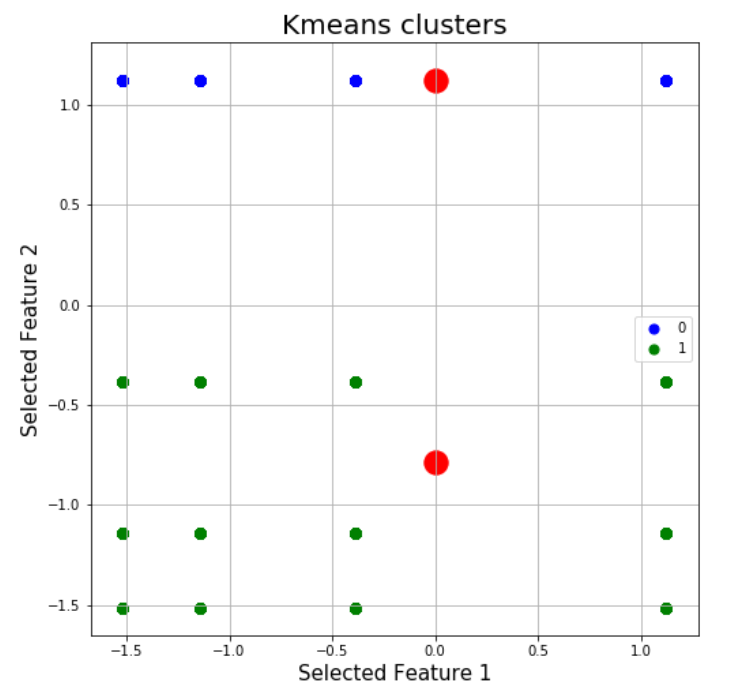
From the plots for ICA we can see the data points are more compressed in comparison to PCA. The RCA plot show the data is more spread out than PCA and it is also evenly distributed among both classes.

From all the above plots we can see that our dataset does not appear to be linearly separable. Which suggests that k means and EM algorithms will not be able to explain a lot of variation in the dataset.

**Experiment 3:**

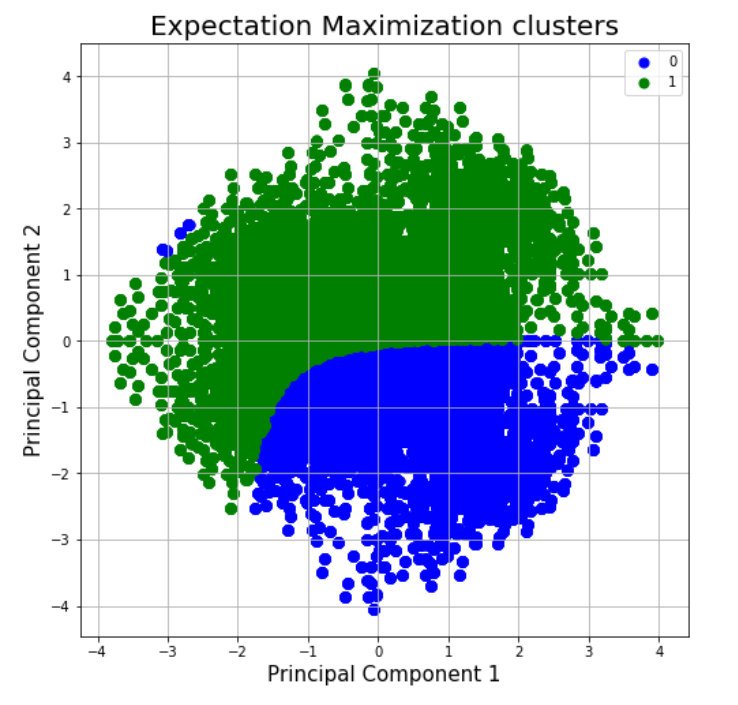
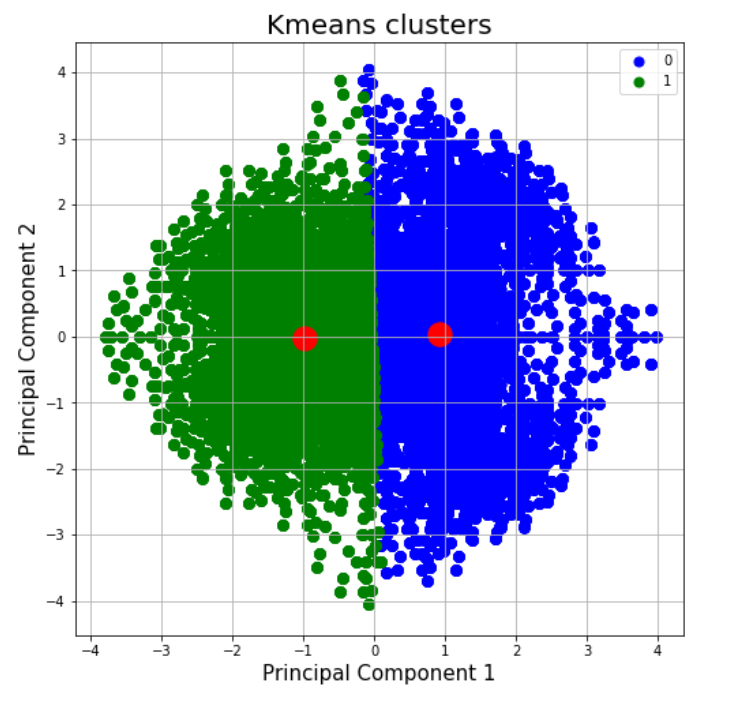
In this experiment we will be running k means and EM algorithms on the dimensionally reduced dataset from the previous experiment. We will be plotting the resulting clusters to analyze the performance of the algorithms. We will also be using accuracy as a metric to compare their relative performances.

Feature Selection:



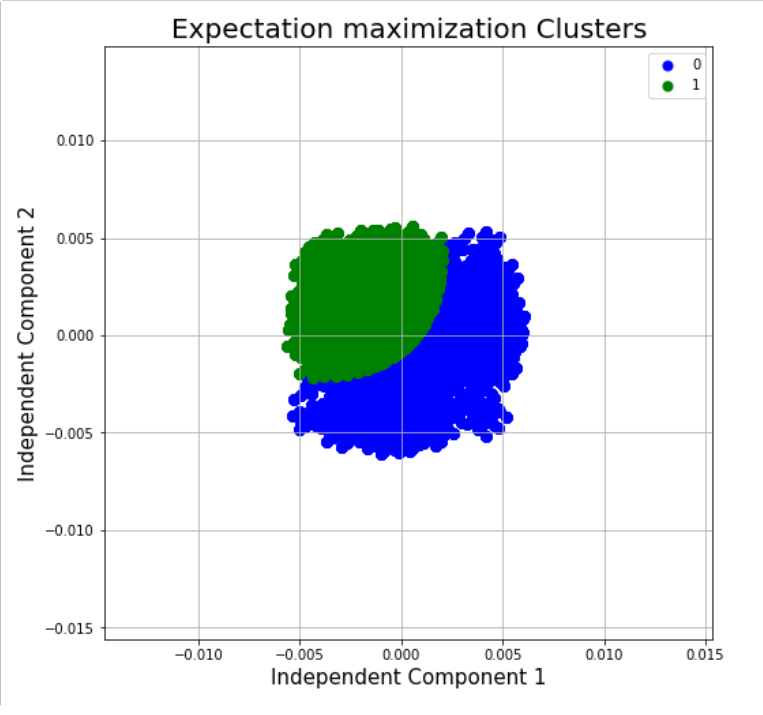
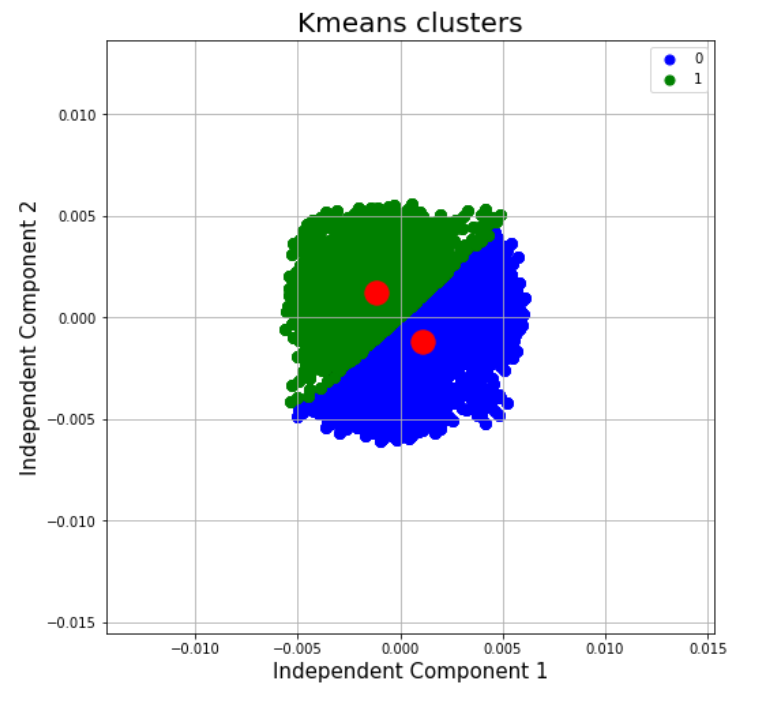
Comparing the above plots to the plot for feature selection from experiment 2, we can conclude that k means and EM do not do a good job of isolating the 2 classes. Since the accuracy of the k means model is 37.48% and EM model is 69.84% this confirms our inference.

PCA:



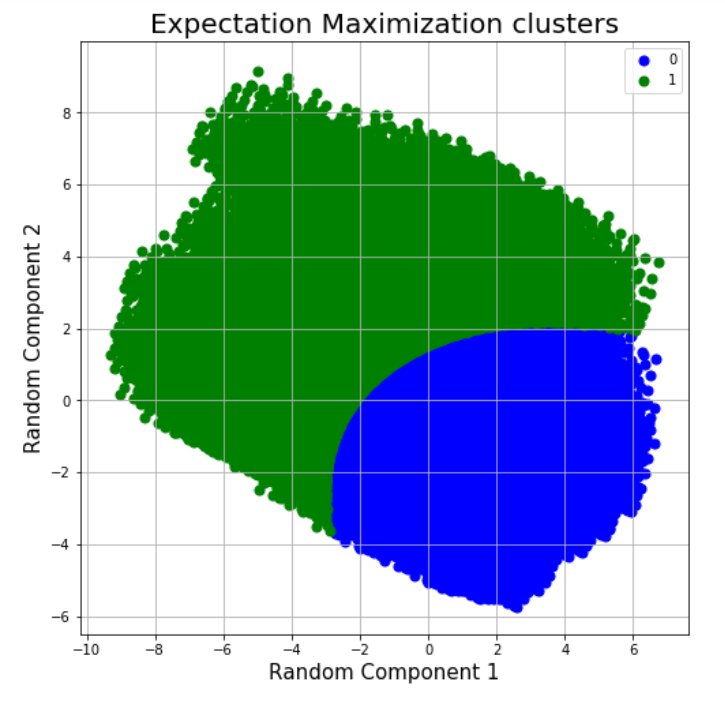
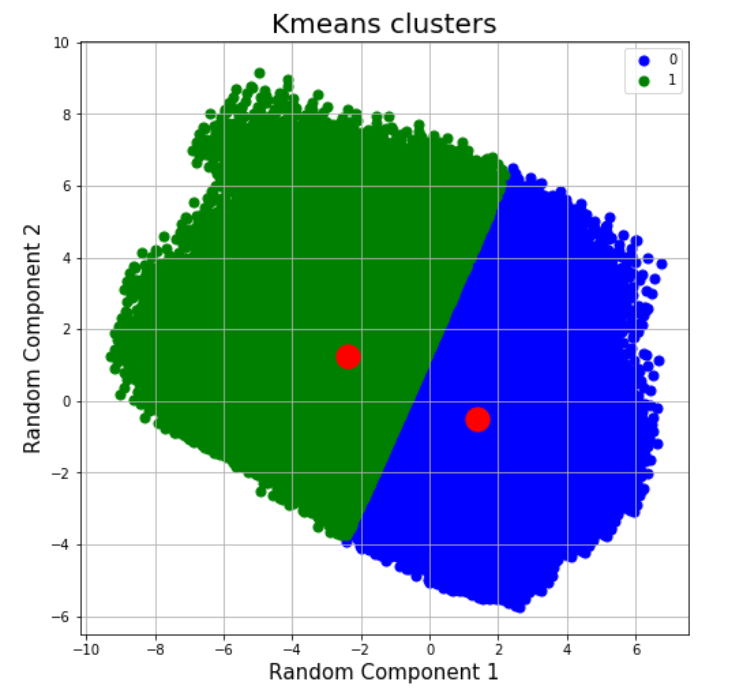
Comparing these plots with the plot of the dataset from experiment 2, we can see that these models can only around half of the variation in the data. Since the model accuracies are 41.55% and 50.57% this supports our inference.

ICA:



Looking at the plots we can see that the perform of our models will be similar to PCA. We can also confirm this by looking at the model accuracies which are 41.54% and 42.35% respectively.

RCA:

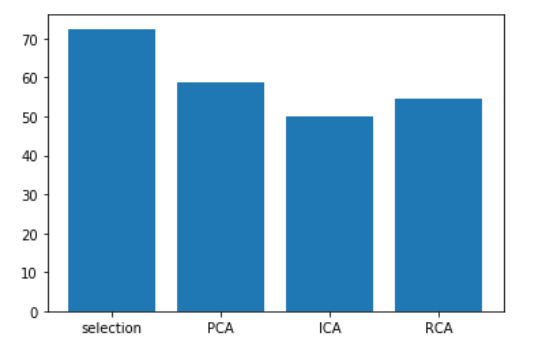


The above plots indicate that the performance of these models will be similar to the PCA and ICA models. We can confirm the same from the accuracies of the models which are 51.89% and 54.02% respectively.

From the above metric we can see that EM performs better than K means for this dataset.

**Experiment 4:**

For this experiment we will using the same dimensionality reduction techniques from the above experiments and we will be observing their impact on a neural network. The neural network we will be using for this study will have one hidden layer with 128 nodes and one output layer with 2 nodes, all running sigmoid activation functions. Running a model using this neural network on the dimensionally reduced dataset from the previous experiments we get the below results:



From the above graph we can see that the model with feature selection performance between than the other model with an average accuracy of 72.50%.

**Experiment 5:**

For this experiment we will running the same neural network from the previous experiment on the outputs from our k means and EM models from experiment 1. Running this model, we get an average accuracy 70.04% which is better than most of the models from experiment 4, expect for our model with feature selection. So, we can infer that the although the k means and EM model do not help in classifying the data. They provide new information regarding the data that can be used to boost the performance of neural networks.