APPLIANCES ENERGY PREDICTION & BANK TERM LOAN DEPOSIT PREDICTION USING CLUSTERING ALGORITHMS

SUDHARSANA RAJASEKARAN

APPLIED MACHINE LEARNING

Assignment 4

Table of Contents

K-Means Clustering

Finding the Optimal Number of Clusters

Determining the optimal number is cluster is essential in a clustering problem in towards solving it. Here we ran the elbow plot to determine the optimal number of clusters in the dataset. In the first dataset, the elbow is not clearly visible, hence the optimal number of clusters can be 2,3 or 4. Hence to conclude the optimal number of clusters we need to use other optimal cluster finding the methods. In dataset 2, it is clearly visible that the elbow occurs at 2, hence the optimal number of clusters, in this case, is 2.

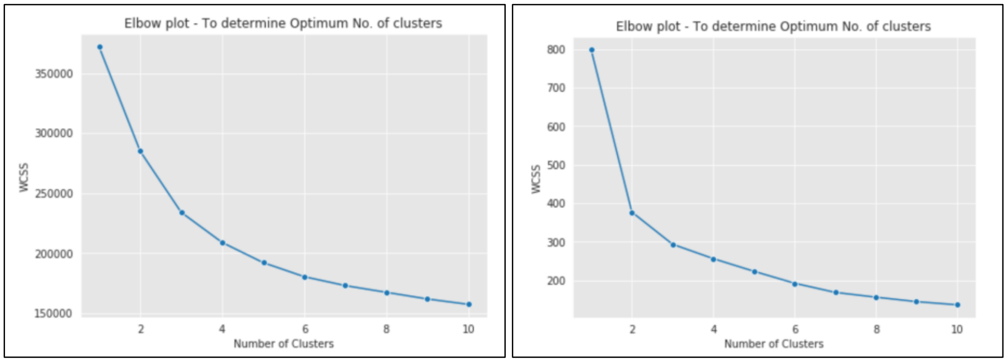


Figure 1. Finding the Optimal number of clusters using Elbow plot for both the datasets

Silhouette plot is a clustering finding method to find the optimal number of clusters. Here from the below graph, the maximum silhouette scores occur for a cluster of 2 in both the datasets. Therefore, it can be concluded that the number of clusters for maximum output performance is 2 in both cases. This cluster of 2 aligns with the number of class labels in the dataset.

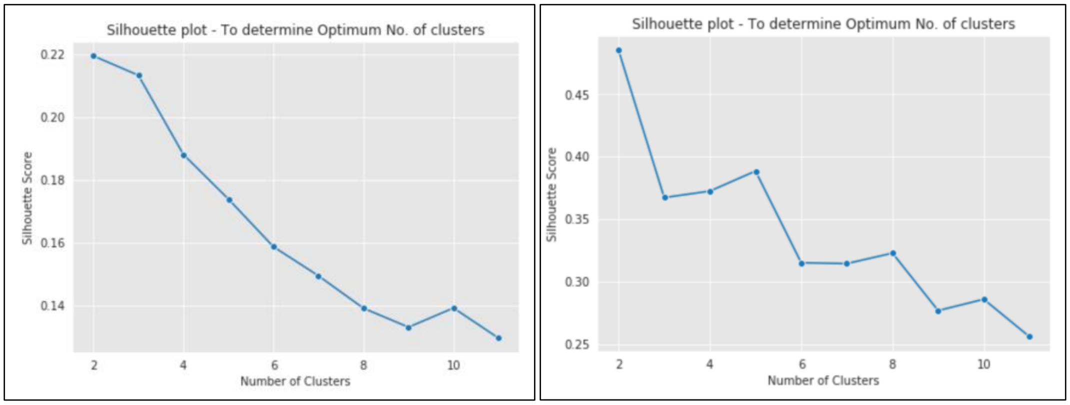
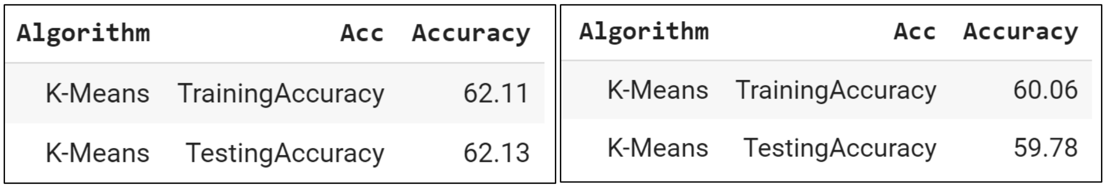


Figure 2. Finding the Optimal number of clusters using Silhouette plot for both the datasets

Training and Test Accuracies

The K-Means algorithm is run for 2 clusters and we got a train and test accuracy of 62% on the first dataset and around 60% on the second dataset. The train and test accuracies are close to each other, meaning the model can generalize well on the test dataset. This is confirming the correctness of the number of clusters we chose for the datasets for the classification problem.



Algorithm	Acc	Accuracy
K-Means	TrainingAccuracy	62.11
K-Means	TestingAccuracy	62.13

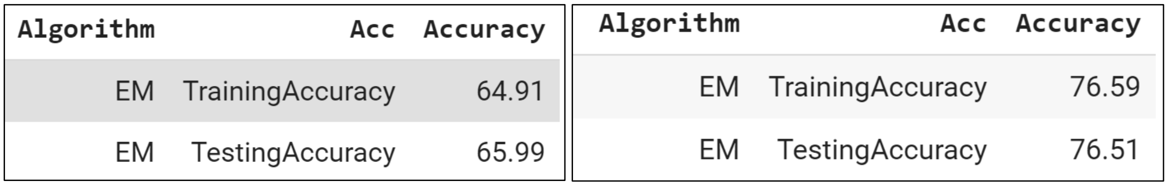
Algorithm	Acc	Accuracy
K-Means	TrainingAccuracy	60.06
K-Means	TestingAccuracy	59.78

Figure 3.Test and Train Accuracies of K-Means algorithm for both the datasets

Expectation Maximization

Gaussian model with EM Algorithm

Gaussian Mixture Models are a powerful clustering algorithm. Expectation-Maximization (EM) is a statistical algorithm for finding the right model parameters. We typically use EM when the data has missing values, or in other words, when the data is incomplete. By using a Gaussian mixture model with the Expectation-Maximization algorithm we were to improve the performance of the model. We were able to get a test and train accuracies of around 65% in the first dataset and around 76% in the second dataset. The test and train accuracies being close by indicating the models that were built are being able to generalize well on the test dataset.



Algorithm	Acc	Accuracy
EM	TrainingAccuracy	64.91
EM	TestingAccuracy	65.99

Algorithm	Acc	Accuracy
EM	TrainingAccuracy	76.59
EM	TestingAccuracy	76.51

Figure 4.Test and Train Accuracies of EM algorithm for both the datasets

Comparison of K-Means and Expectation Maximization Algorithm

On Comparing the performance of K-Means and EM algorithm, we can see that the Expectation-Maximization algorithm outperforms the K-means algorithm in both the datasets. We can achieve a test and train accuracy of 62% and 60% on K-Means and a whopping 65% and 75% with Expectation maximization algorithm.

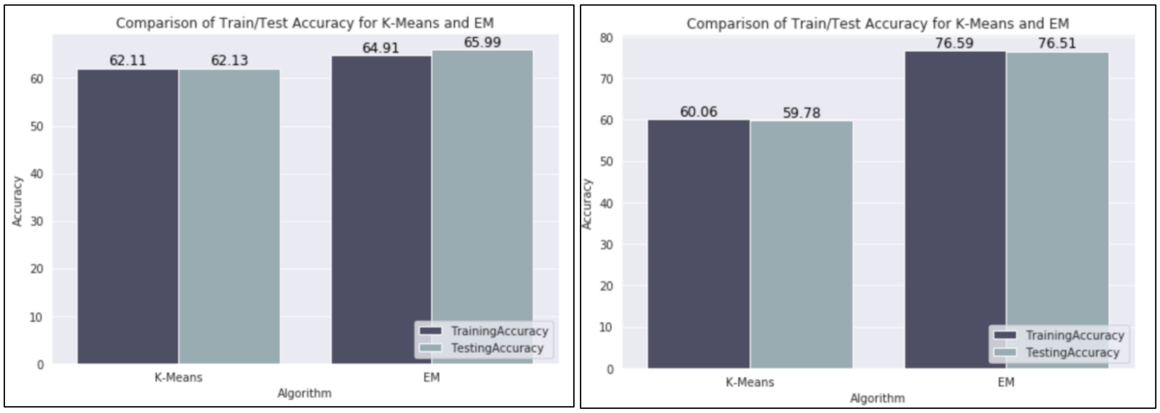
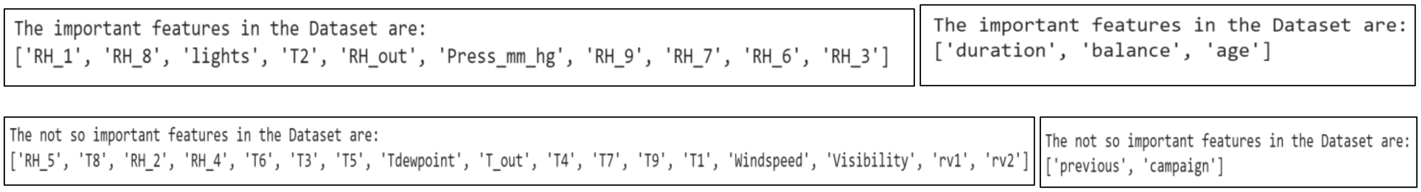


Figure 5.Comparison of performance of K-Means and EM for both the datasets

Dimensionality Reduction Algorithms

Feature Selection

Feature selection is an important dimensionality reduction algorithm, that helps to filter all the unnecessary variables in a dataset. Here, we have used Random Forest classifier as our feature selection algorithm. Using Random forest, we were able to classify the important and unimportant variables in the datasets provided. Hence, we use the important variables to build our model.



```
The important features in the Dataset are:
['RH_1', 'RH_8', 'lights', 'T2', 'RH_out', 'Press_mm_hg', 'RH_9', 'RH_7', 'RH_6', 'RH_3']
```

```
The important features in the Dataset are:
['duration', 'balance', 'age']
```

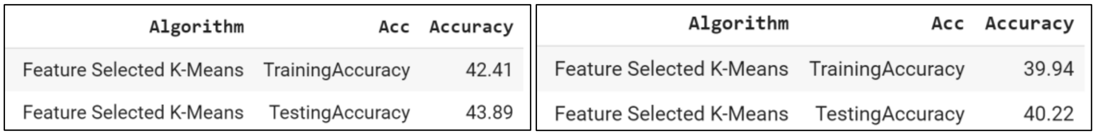
```
The not so important features in the Dataset are:
['RH_5', 'T8', 'RH_2', 'RH_4', 'T6', 'T3', 'T5', 'Tdewpoint', 'T_out', 'T4', 'T7', 'T9', 'T1', 'Windspeed', 'Visibility', 'rv1', 'rv2']
```

```
The not so important features in the Dataset are:
['previous', 'campaign']
```

Figure 6.Feature Selection for both the datasets

Feature Selected K-Means

Using the Feature selected K-Means variables from the Random Forest algorithm we were able to model a K-Means algorithm with only the features selected variables final model has a model performance of around 40% in both the datasets. The model built can generalize well in both the datasets.



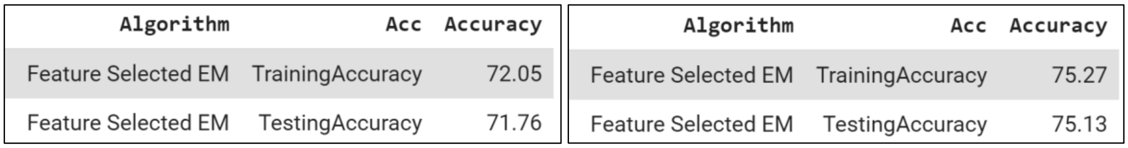
Algorithm	Acc	Accuracy
Feature Selected K-Means	TrainingAccuracy	42.41
Feature Selected K-Means	TestingAccuracy	43.89

Algorithm	Acc	Accuracy
Feature Selected K-Means	TrainingAccuracy	39.94
Feature Selected K-Means	TestingAccuracy	40.22

Figure 7.Train and Test accuracies of K-Means for both the datasets

Feature Selected EM

EM algorithm that built on top of the feature selected variables was performing better than Feature selected K-Means algorithm. Feature selected EM was able to achieve a train and test accuracies of 72% in the first dataset and 75% in the second dataset. There is consistency in the train and test accuracy. Hence, we are can conclude that the models can generalize well.



Algorithm	Acc	Accuracy
Feature Selected EM	TrainingAccuracy	72.05
Feature Selected EM	TestingAccuracy	71.76

Algorithm	Acc	Accuracy
Feature Selected EM	TrainingAccuracy	75.27
Feature Selected EM	TestingAccuracy	75.13

Figure 8.Train and Test accuracies of EM for both the datasets

Comparison of performance of Feature selected K-Means and EM algorithm

On comparison, we can see that the Feature selected K-Means performed poorly than K-Means with all the variables. On the other hand, the feature selected EM algorithm outperformed the EM algorithm with all the variables.

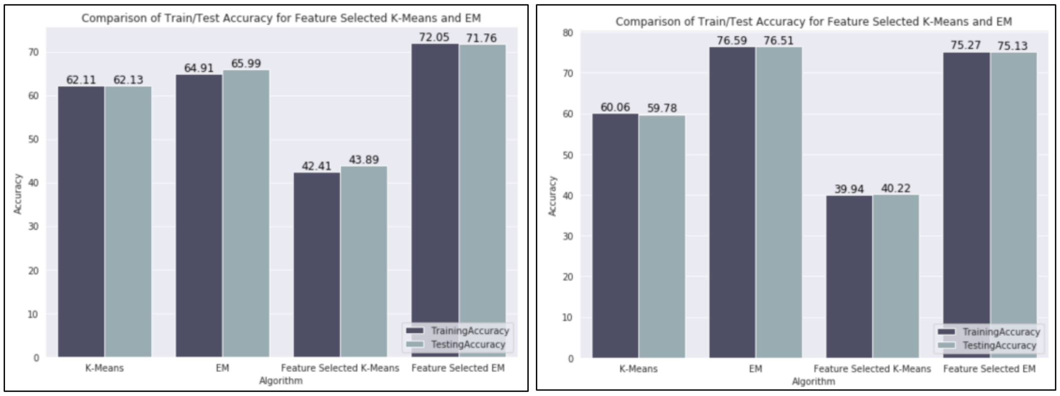


Figure 9.Comparison of performance of K-Means and EM for both the datasets

Principal Component Analysis

Explained Variance

The principal component analysis is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. Before the PCA algorithm could be run, tested on a number of components to be retained for a maximum variance. We then pass on the not so important features into PCA algorithm and 8 components from PCA has to be retained for a minimum 95% explained variance and for the Dataset 2, only one PCA component is enough to explain a 95% variance.

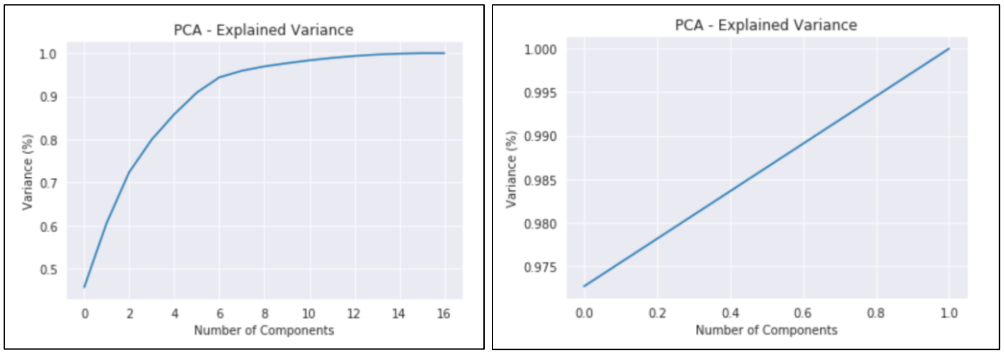


Figure 10.PCA – Explained Variance for both the datasets

Silhouette Plot

After PCA we run the silhouette plot to find the optimal number of clusters before running the K-Means algorithm. It can be observed that even after PCA the optimal number of clusters seems to remain at 2.

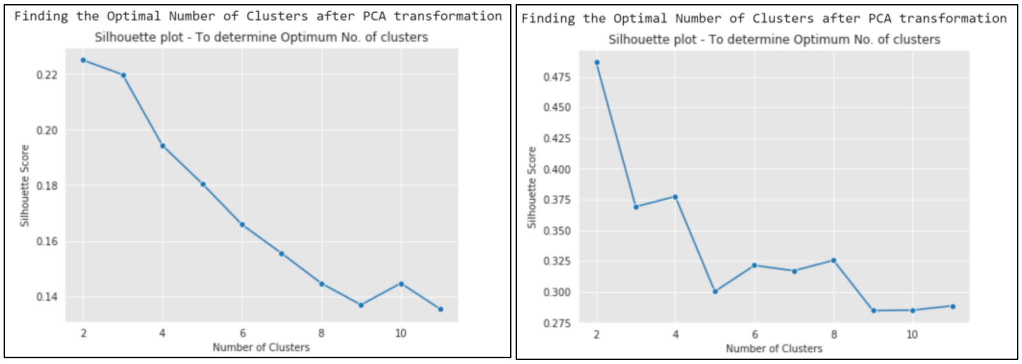
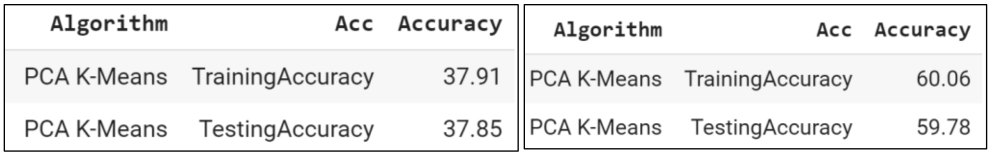


Figure 11.Silhouette plot after PCA for both the datasets

K-Means

Running the K-Means algorithm after PCA resulted in a training and test accuracies of 37% in the first dataset and approximately 60% in the second dataset.



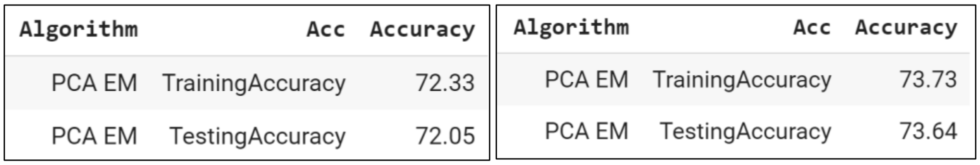
Algorithm	Acc	Accuracy
PCA K-Means	TrainingAccuracy	37.91
PCA K-Means	TestingAccuracy	37.85

Algorithm	Acc	Accuracy
PCA K-Means	TrainingAccuracy	60.06
PCA K-Means	TestingAccuracy	59.78

Figure 12.Train and Test accuracies of K-Means (PCA) for both the datasets

EM

Running EM algorithm after PCA resulted in a training and test accuracies of 72% in the first dataset and approximately 73% in the second dataset.



Algorithm	Acc	Accuracy
PCA EM	TrainingAccuracy	72.33
PCA EM	TestingAccuracy	72.05

Algorithm	Acc	Accuracy
PCA EM	TrainingAccuracy	73.73
PCA EM	TestingAccuracy	73.64

Figure 13. Train and Test accuracies of EM (PCA) for both the datasets

Comparison of PCA transformed, feature selected K-Means and EM Algorithms

On comparison, PCA transformed EM algorithm outperforms all others in the first dataset and EM algorithm outperforms all other algorithms in the second dataset.

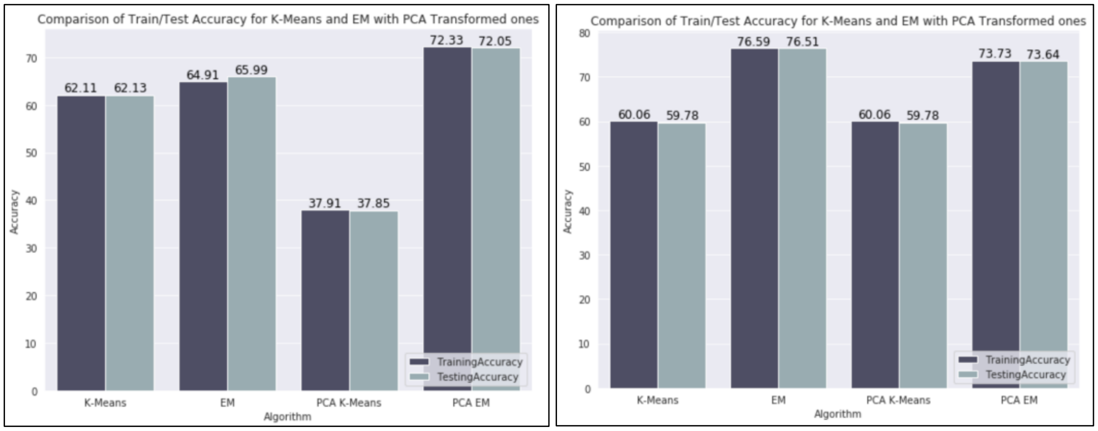


Figure 14.Comparison of performance of K-Means and EM (PCA) for both the datasets

Independent Component Analysis

Silhouette Plot

After applying ICA, we tried to find the optimal number of clusters with which we can run the silhouette plot to find. From the silhouette plot we can find the optimal number of clusters is 2 for both the datasets after ICA.

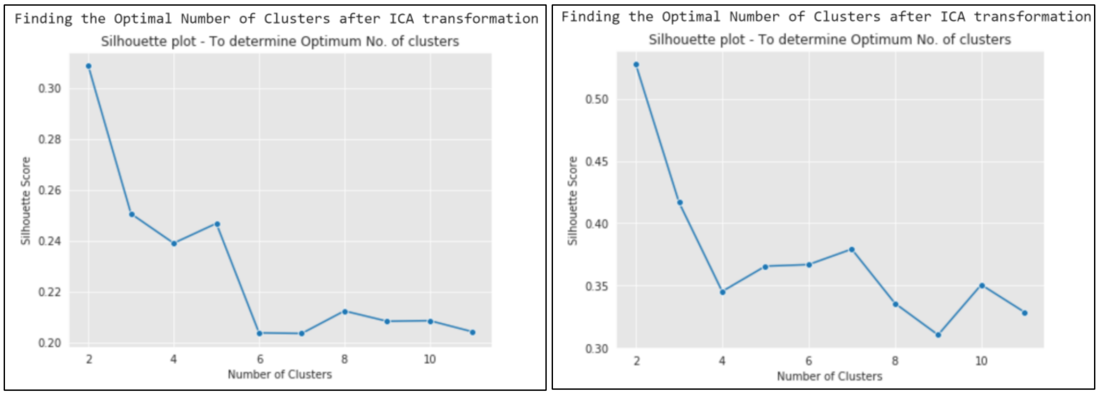
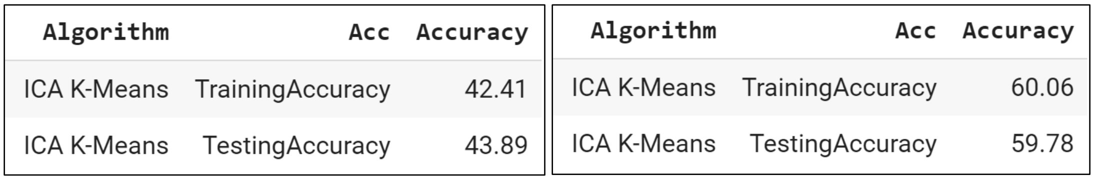


Figure 15. Silhouette plot after ICA for both the datasets

K-Means

Running K-Means algorithm after ICA resulted in a training and test accuracies of 43% in the first dataset and approximately 60% in the second dataset.



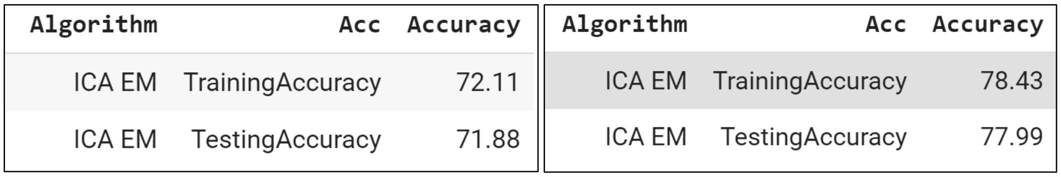
Algorithm	Acc	Accuracy
ICA K-Means	TrainingAccuracy	42.41
ICA K-Means	TestingAccuracy	43.89

Algorithm	Acc	Accuracy
ICA K-Means	TrainingAccuracy	60.06
ICA K-Means	TestingAccuracy	59.78

Figure 16. Train and Test accuracies of K-Means (ICA) for both the datasets

EM

Running EM algorithm after ICA resulted in a training and test accuracies of 43% in the first dataset and approximately 60% in the second dataset.



Algorithm	Acc	Accuracy
ICA EM	TrainingAccuracy	72.11
ICA EM	TestingAccuracy	71.88

Algorithm	Acc	Accuracy
ICA EM	TrainingAccuracy	78.43
ICA EM	TestingAccuracy	77.99

Figure 17. Train and Test accuracies of EM (ICA) for both the datasets

Comparison of ICA transformed, feature selected K-Means and EM Algorithms

It can be observed that the Feature selected, ICA transformed EM algorithm outperformed all others in both the datasets.

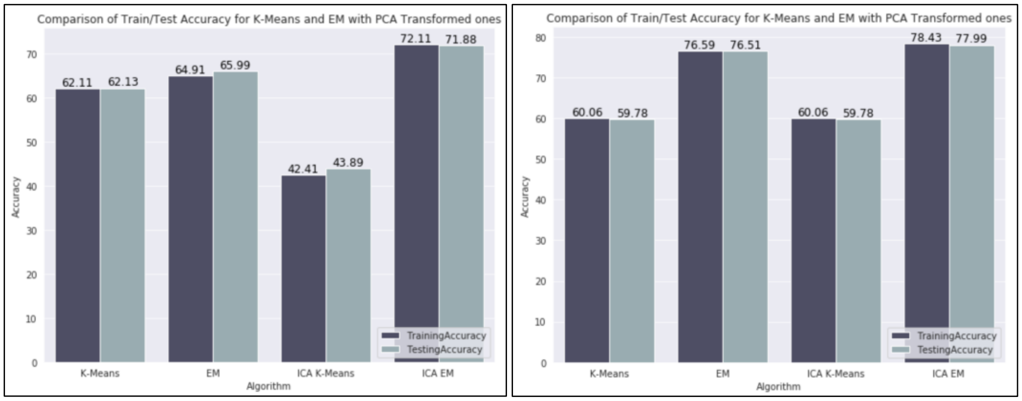


Figure 18.Comparison of performance of K-Means and EM (ICA) for both the datasets

Randomized Projections

Silhouette Plot

After applying RP, we tried to find the optimal number of clusters with which we can run the silhouette plot to find. From the silhouette plot we can find the optimal number of clusters is 2 for first dataset and 3 for the second dataset.

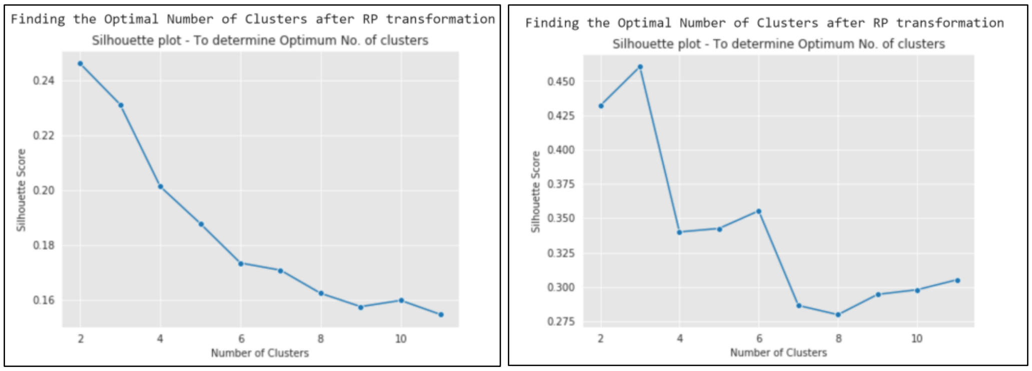
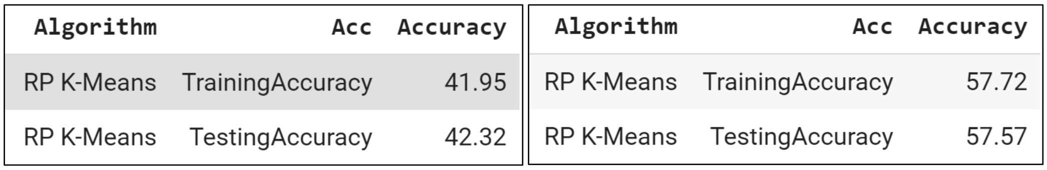


Figure 19. Silhouette plot after RP for both the datasets

K-Means

Running K-Means algorithm after RP resulted in a training and test accuracies of 43% in the first dataset and approximately 57% in the second dataset.



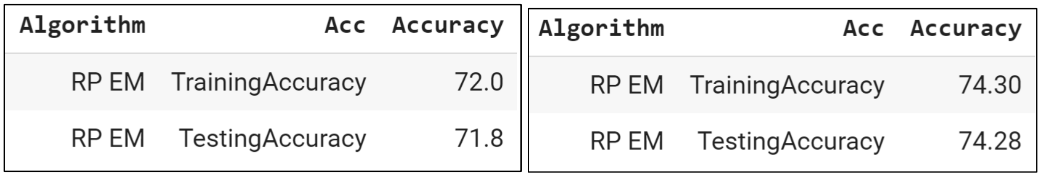
Algorithm	Acc	Accuracy
RP K-Means	TrainingAccuracy	41.95
RP K-Means	TestingAccuracy	42.32

Algorithm	Acc	Accuracy
RP K-Means	TrainingAccuracy	57.72
RP K-Means	TestingAccuracy	57.57

Figure 20. Train and Test accuracies of K-Means (RP) for both the datasets

EM

Running K-Means algorithm after ICA resulted in a training and test accuracies of 72% in the first dataset and approximately 74% in the second dataset.



Algorithm	Acc	Accuracy
RP EM	TrainingAccuracy	72.0
RP EM	TestingAccuracy	71.8

Algorithm	Acc	Accuracy
RP EM	TrainingAccuracy	74.30
RP EM	TestingAccuracy	74.28

Figure 21. Train and Test accuracies of EM (RP) for both the datasets

Comparison of RP transformed, feature selected K-Means and EM Algorithms

It can be observed that the feature selected, RP transformed EM algorithm outperformed all the others on the first dataset and EM algorithm outperformed all others in the second dataset.

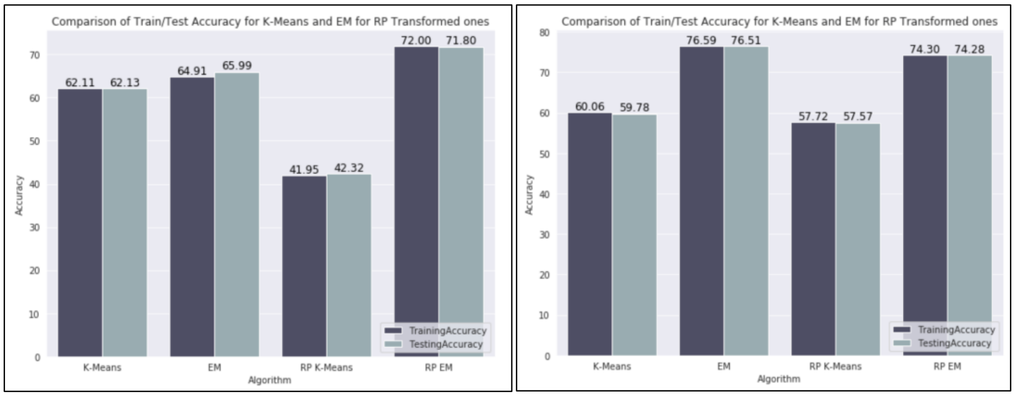
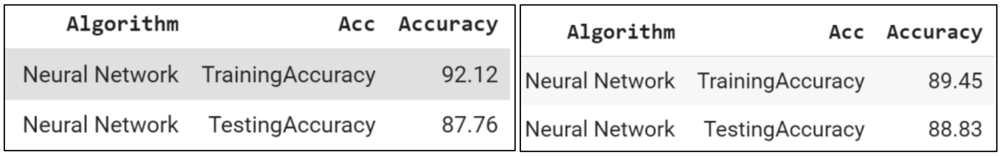


Figure 22.Comparison of performance of K-Means and EM (RP) for both the datasets

Neural Network

On running a Neural Network with 3 hidden layers, we were able to achieve a train and test accuracies of 92% and 89% in both the datasets.



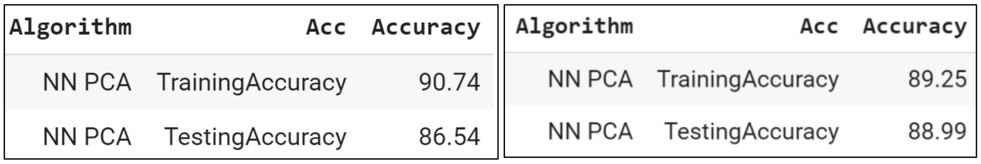
Algorithm	Acc	Accuracy
Neural Network	TrainingAccuracy	92.12
Neural Network	TestingAccuracy	87.76

Algorithm	Acc	Accuracy
Neural Network	TrainingAccuracy	89.45
Neural Network	TestingAccuracy	88.83

Figure 23.Train and test accuracies for Neural Network for both the datasets

PCA transformed Neural Network

With PCA transformed Neural Network, the train and test accuracies are recorded to be 90% and 89% for both the datasets.



Algorithm	Acc	Accuracy
NN PCA	TrainingAccuracy	90.74
NN PCA	TestingAccuracy	86.54

Algorithm	Acc	Accuracy
NN PCA	TrainingAccuracy	89.25
NN PCA	TestingAccuracy	88.99

Figure 24.Train and test accuracies of PCA transformed Neural Network for both the datasets

Performance comparison of Neural Network models

On comparison, even though the accuracy of PCA transformed Neural Network was slightly lower than the Neural network. This is small degradation performance is acceptable, given small training time.

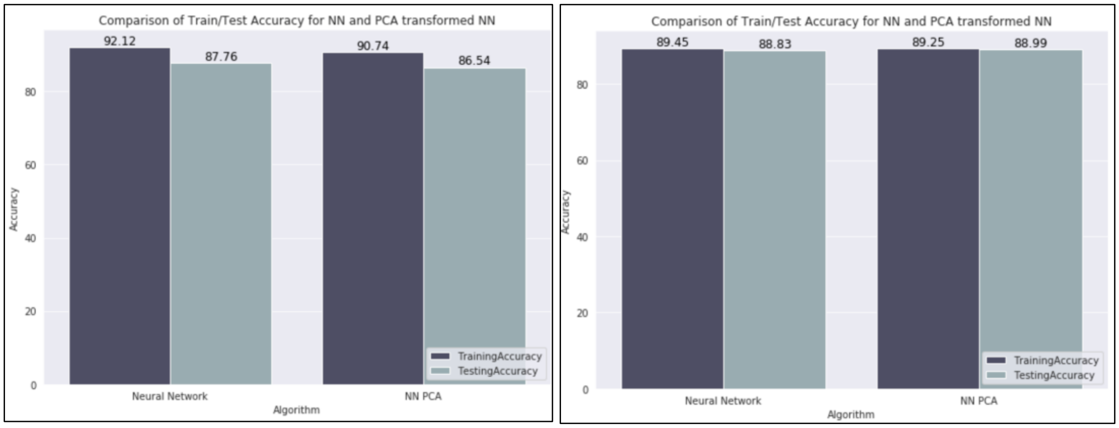
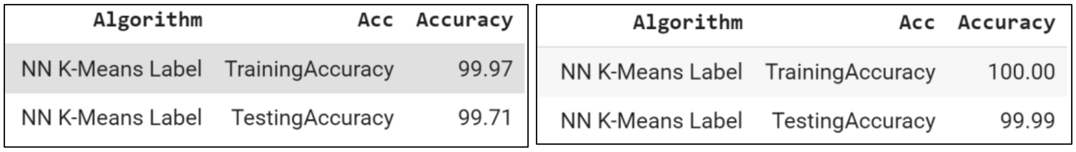


Figure 25.Comparison of performance of Neural Network (NN) and PCA transformed NN for both the datasets

Neural Network with label generated by K-Means

Instead of classification labels in the dataset, if we were using the labels generated by K-Means to predict the output by using a neural network, we can achieve a whooping accuracy of 99%. This is because of k-means label clusters based on patterns in the dataset. This pattern is then easily picked up the neural network for its classification.



Algorithm	Acc	Accuracy
NN K-Means Label	TrainingAccuracy	99.97
NN K-Means Label	TestingAccuracy	99.71

Algorithm	Acc	Accuracy
NN K-Means Label	TrainingAccuracy	100.00
NN K-Means Label	TestingAccuracy	99.99

Figure 26.Comparison of performance of NN with label generated by K-Means for both the datasets

Comparison of Neural Network Models

On comparison, it can be observed that the Neural network with label generated by K-Means performed better than the other. This is because of k-means label clusters based on patterns in the dataset. This pattern is then easily picked up the neural network for its classification.

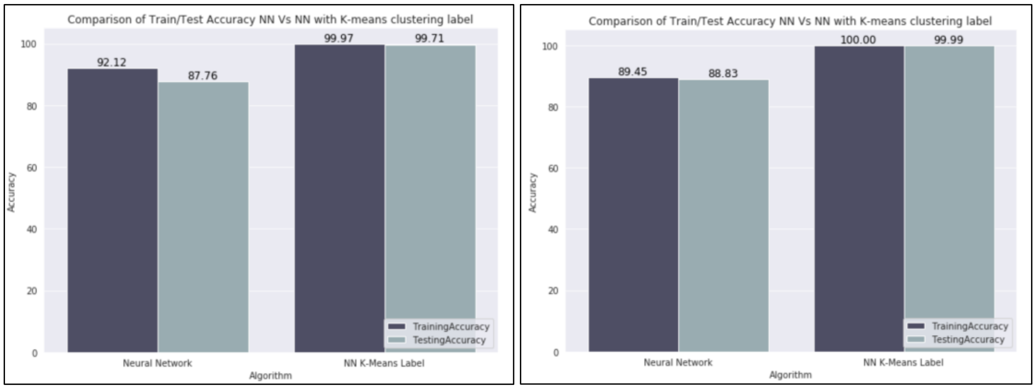


Figure 27.Comparison of performance of Neural Network models for both the datasets

Comparison of all the models

Finally, on comparison of all models, Neural networks with label generated by K-means outperformed all the models in both the datasets. This is because of k-means label clusters based on patterns in the dataset. This pattern is then easily picked up the neural network for its classification.

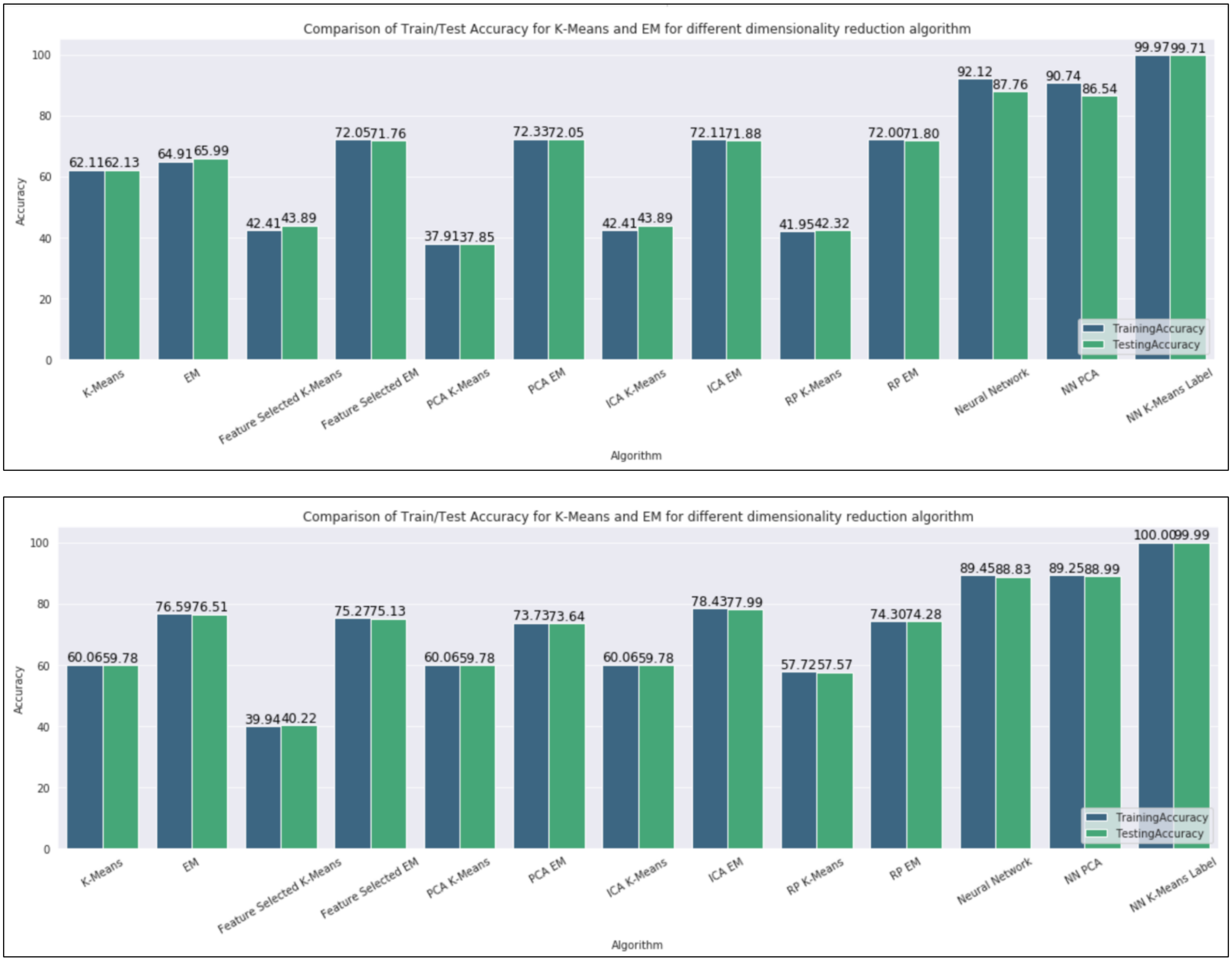


Figure 28.Comparison of performance of all models for both the datasets

Appendix

DATASET 1 – APPLIANCE ENERGY PREDICTION

DATASET 2 – BANK LOAN TERM DEPOSIT PREDICTION