SEOUL BIKE DATA DEMAND

INTRODUCTION

Public bike-sharing systems have been gaining momentum only in the last decade. The main purpose other than convenience and easy-to-use service for customers is the mobility. More people are turning to healthier life styles and locations where bike riding can be easily available. There are many benefits in bike riding. Therefore, it is important to have rental bikes available to the customers (in our case, the public) to reduce their waiting time.

In this project, we implement a binary classification problem. It helps to give us the best model selected through experimentation and then evaluate the model for prediction, which is to predict the rental bike demand.

DATASET

This dataset has 8760 records comprising the details of every hour each day and also 14 columns. The data contains the hourly and daily count of rental bikes. It also contains the weather information (Rainfall, Snowfall, Temperature, Humidity, Visibility etc). Rented Bike Count is the current dependent variable when checked in the dataset.

We plan to implement different clustering algorithms – kMeans and Expectation Maximization and ran Artificial Neural Network machine algorithm to see if our model generalises well. Also, feature Selection techniques using RF feature importance and feature transformation techniques like Principal Component Analysis (PCA), Independent Component Analysis (ICA) and Random Projections are performed.

DATA CLASSIFICATION PROBLEM

The dependent variable is Rented Bike Count which is a numerical variable. We will change our dependent variable to a binary classification problem by thresholding. Current threshold taken for our analysis is 75th percentile. A good reason for creating binary predictors from numerical predictors is to overcome the problem of linearity. According to linear regression, we assume that X and Y to be in linear relation. If we can’t find an appropriate equation to represent relation between X and Y variables , then creating binary predictors might be a way of obtaining some predictors .

The dataset problem is such that to see if we can predict if the bikes rented for each hour is greater than the threshold (75th percentile). This problem also takes into account the weather conditions to determine if they have an impact on prediction. For this, EDA has been conducted and we could see that positive correlation was seen for Temperature and Dew Point Temperature. Also, fields like Seasons showed Summer as the peak season for rented bike counts and also an increase in demand of bike count for higher temperature.

ALGORITHMS

1. ARITIFICIAL NEURAL NETWORKS

Artificial Neural Networks or Neural Networks is an efficient computing system where the main concept is borrowed from the analogy of biological neural networks. Multiple input signals, referred as Input Layer Neurons, are transmitted to Hidden Layer Neurons, which on its turn are used to predict the output, Output Layer. In the sense, input signals are similar to the human senses, such as human sight, hearing, smelling, tasting and touching, only in the case of ANN, those input signals can be various type of features characterizing an observation

ANN acquires a large collection of units that are interconnected in some pattern to allow communication between the units. These units, also referred to as nodes or neurons, are simple processors which operate in parallel. Every neuron is connected with other neuron through a connection link. Each connection link is associated with a weight that has information about the input signal. This is the most useful information for neurons to solve a particular problem because the weight usually excites or inhibits the signal that is being communicated. Each neuron has an internal state, which is called an activation signal. Output signals, which are produced after combining the input signals and activation rule, may be sent to other units.

Data Normalization

Artificial Neural Networks are sensitive to the scale of the data. Therefore, we will use the Standard Scalar from the Scikit learn library to scale the data.Given that the dependent variable that can only take values 0 and 1, we only need to scale the features data, the set of all independent variables.

Data Splitting

In order to make sure we first train the model using only a part of the data and then use the trained model to predict the rented bike count whose data have not been used during the training.

Activation Function

We use a function to join signals from different input neurons into one value. Each synapse get assigned a weight, an importance value. These weights form the corner stone of how Neural Networks learn. These weights determine whether the signals get passed along or not, or to what extent each signal gets passed along. If we define the input value of signal i by xi and its importance weight by wi then the sum of these signals, can be defined by

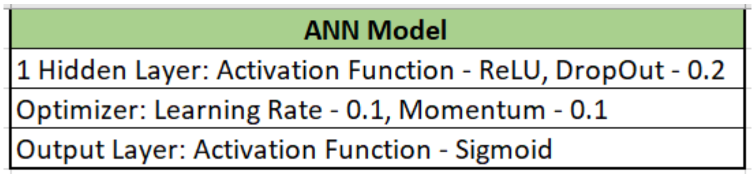
$$\phi(\sum_{i=1}^{m} w_i x_i)$$

where φ represents the activation function. This function is called Activation Function. We have used sigmoid, tanh and relu activation functions for our analysis.

MODEL TUNING

For our classification problem, we created a model algorithm with the below layers (as shown in Fig 1 .1) and size and we got a good accuracy.

Neural Network have a lot of parameters to find the optimum parameters with the best scores. We have used cost function optimizers like SGD (Stochastic Gradient Decent). It is an iterative approach for solving optimization problems with a different function. It aims to find the extreme or zero points of the stochastic model containing parameters that cannot be directly estimated. Post this, we will use Sigmoid activation function because our problem is a binary classification rented bike count problem. Loss function used is cross entropy which gives the negative average of the log of corrected predicted probabilities. GridSearchCV from SkLearn package has been used for the analysis.



ANN Model
1 Hidden Layer: Activation Function - ReLU, DropOut - 0.2
Optimizer: Learning Rate - 0.1, Momentum - 0.1
Output Layer: Activation Function - Sigmoid

Fig 1.1 ANN layer network

2. K-MEANS

K-means algorithm is defined as an supervised learning method with an iterative approach where dataset is grouped into k number of clusters. This iterative algorithm partitions the dataset according to their features into K number of predefined non- overlapping distinct clusters or subgroups. It allocates the data points to a cluster if the sum of the squared distance between the cluster's centroid and the data points is at a minimum.

ELBOW METHOD

This is a curve drawn between "within the sum of squares" (WSS) and the number of clusters. It is called the elbow method because the point of the elbow in the curve gives us the optimum number of clusters. In the graph or curve, after the elbow point, the value of WSS chang es very slowly, so the elbow point must be considered to give the final value of the number of clusters.

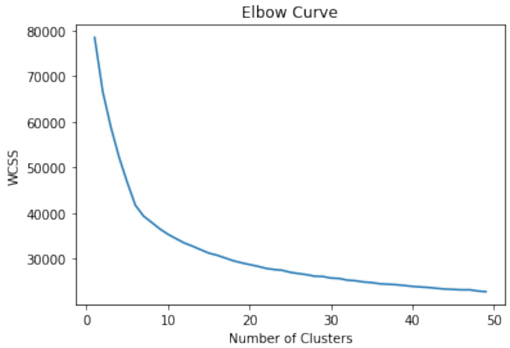
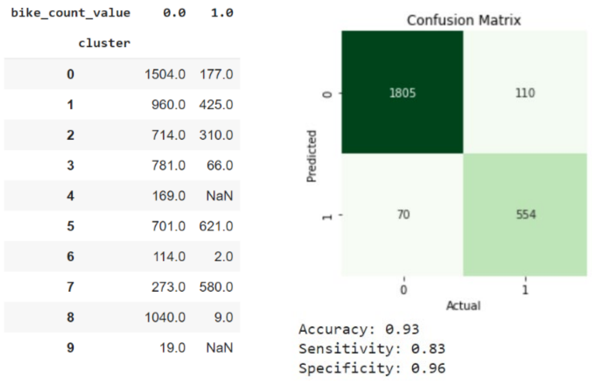


Fig 2.1 Elbow Method

In Fig 2.1, the WCSS decreases as the cluster size increases. We have chosen the optimum cluster as 10 in our case. In Fig 2.3, the box plot of the respective features is with respect to the 10 classes. Almost all distributions show symmetry which means the clusters are good and compact.



bike_count_value	0.0	1.0
cluster		
0	1504.0	177.0
1	960.0	425.0
2	714.0	310.0
3	781.0	66.0
4	169.0	NaN
5	701.0	621.0
6	114.0	2.0
7	273.0	580.0
8	1040.0	9.0
9	19.0	NaN

```
Accuracy: 0.93
Sensitivity: 0.83
Specificity: 0.96
```

Fig 2.2 True/False Predictions as Per Cluster / CM for Test Data

Cluster 5 and 7 shows highest with respect to number of points in the positive class. With respective to this model, we chose the best parameters for number of neurons as 25 and cross validation accuracy as 0.86. Accuracy for test set shows a good value of 0.93.

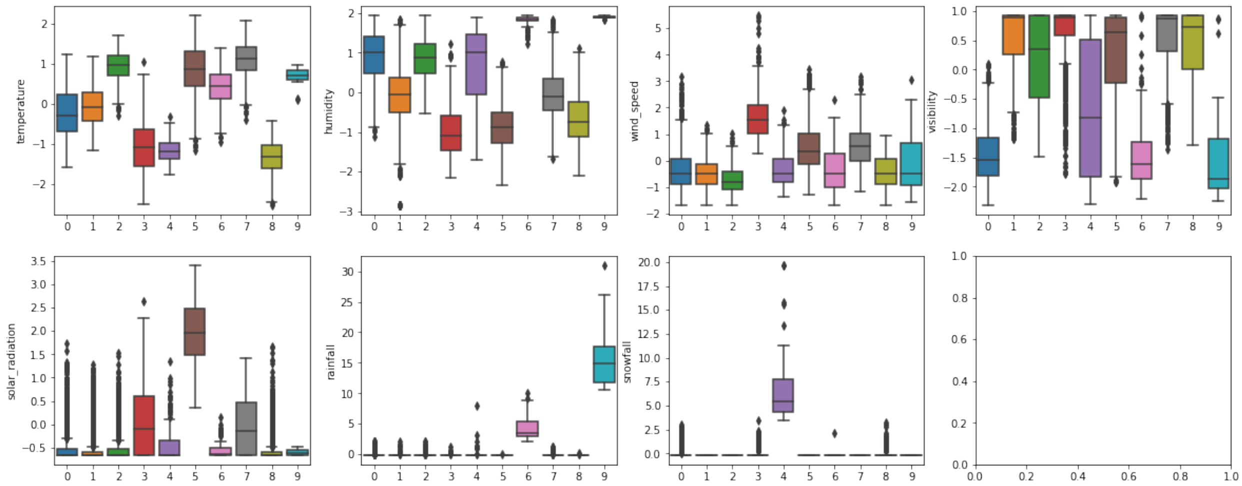


Fig 2.3 Box Plot Distribution - KMeans

3. EXPECTATION MAXIMIZATION

The expectation-maximization algorithm is an approach for performing maximum likelihood estimation in the presence of latent variables. It does this by first estimating the values for the latent variables, then optimizing the model, then repeating these two steps until convergence. It is an effective and general approach and is most commonly used for density estimation with missing data, such as clustering algorithms like the Gaussian Mixture Model. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step.

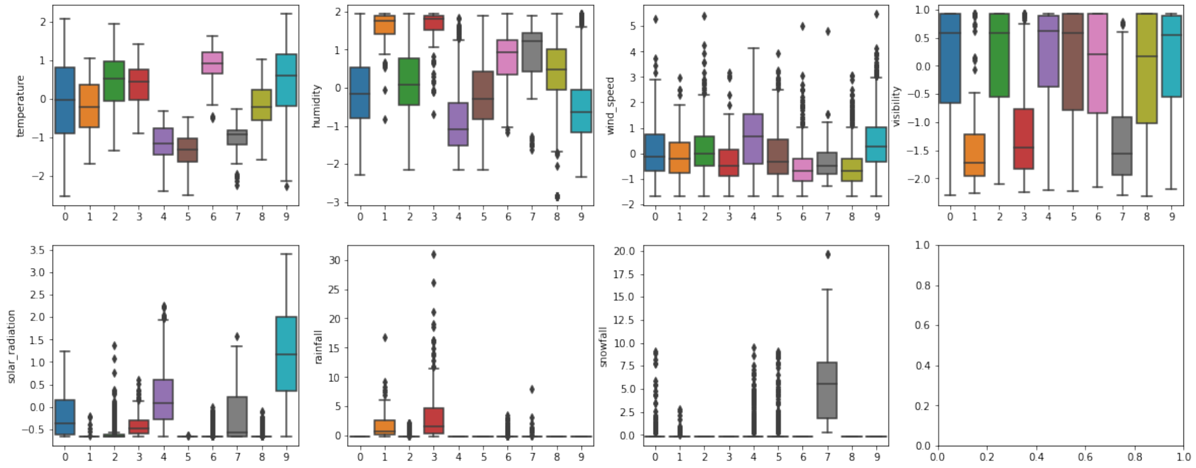
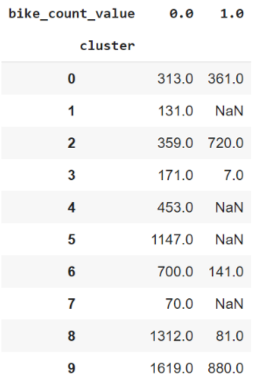
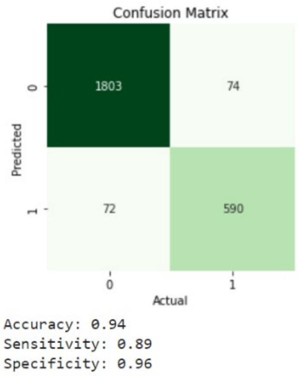


Fig 3.1 Box Plot Distribution – Expectation Maximization



bike_count_value	0.0	1.0
cluster		
0	313.0	361.0
1	131.0	NaN
2	359.0	720.0
3	171.0	7.0
4	453.0	NaN
5	1147.0	NaN
6	700.0	141.0
7	70.0	NaN
8	1312.0	81.0
9	1619.0	880.0



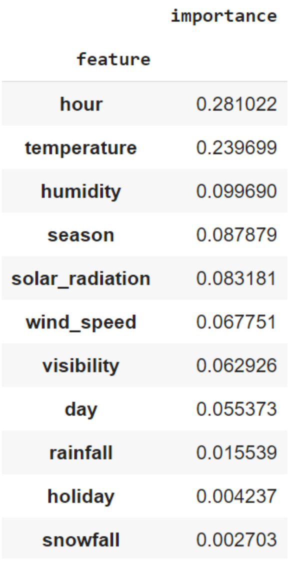
```
Accuracy: 0.94
Sensitivity: 0.89
Specificity: 0.96
```

We can see that rainfall and snowfall are showing distribution such that they have not favourable conditions to rent the bike. With respective to this model, we chose the best parameters for number of neurons as 30 and cross validation accuracy as 0.88. Accuracy for test set shows a good value of 0.94. This result is almost similar to the K-Means result.

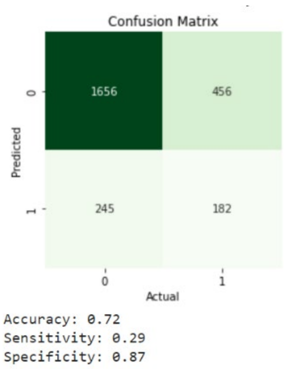
DIMENSION REDUCTION TECHNIQUES

1. RANDOM FOREST – FEATURE IMPORTANCE

Feature selection using Random forest comes under the category of Embedded methods. Embedded methods combine the qualities of filter and wrapper methods. They are implemented by algorithms that have their own built-in feature selection methods. The features that are selected at the top of the trees are in general more important than features that are selected at the end nodes of the trees, as generally the top splits lead to bigger information gains. Therefore, it is possible to compute how much each feature decreases the impurity. The more a feature decreases the impurity, the more important the feature is. In random forests, the impurity decrease from each feature can be averaged across trees to determine the final importance of the variable.



	importance
feature	
hour	0.281022
temperature	0.239699
humidity	0.099690
season	0.087879
solar_radiation	0.083181
wind_speed	0.067751
visibility	0.062926
day	0.055373
rainfall	0.015539
holiday	0.004237
snowfall	0.002703



```
Accuracy: 0.72
Sensitivity: 0.29
Specificity: 0.87
```

In Fig 4.1, we can see that the features that have been selected, which showed the most importance is hour, temperature, humidity, season, solar_radiation, visilibility etc. But, the average Cross Validation accuracy is 0.70 for the best chosen parameters for number of neurons 30. The test accuracy has come down to 0.72 which shows that the neural network uses only a subset of data for the analysis which has reduced the accuracy.

Fig 4.1 Feature Importance , CM – RF

2. PRINCIPAL COMPONENT ANALYSIS

Principal Components Analysis (PCA) is an algorithm to transform the columns of a dataset into a new set of features called Principal Components. By doing this, a large chunk of the information across the full dataset is effectively compressed in fewer feature columns. This enables dimensionality reduction and ability to visualize the separation of classes or clusters if any. It is not a feature selection technique. Rather, it is a feature combination technique. Because each PC is a weighted additive combination of all the columns in the original dataset. However, the PCs are formed in such a way that the first Principal Component (PC1) explains more variance in original data compared to PC2. Likewise, PC2 explains more than PC3, and so on.

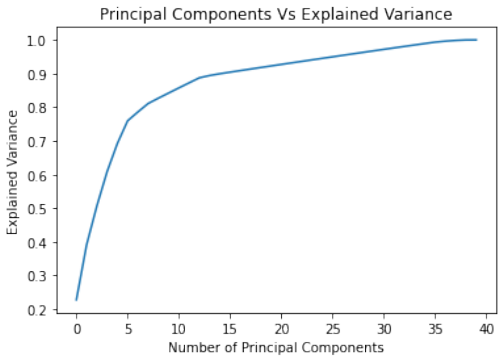
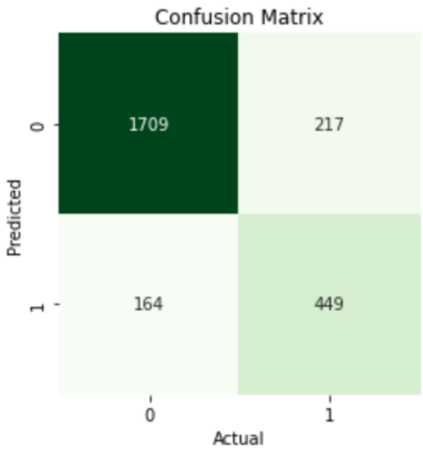


Fig 4.2 EV vs No of PC

In Fig 4.2, this graph shows a summation of the variance with respect to the principal components. We have chosen 13 principal components as the best one as after this, the explained variance graph shows a flat curve.



```
Accuracy: 0.85
Sensitivity: 0.67
Specificity: 0.91
```

The best chosen parameter for number of neurons with 13 features is 15 and cross validation accuracy is 0.77. Test Accuracy is 0.85 which is better than Feature Importance using RF.

From Fig 4.4, the distribution is with respect to first 6 principal components. These separate the points in the binary classes.

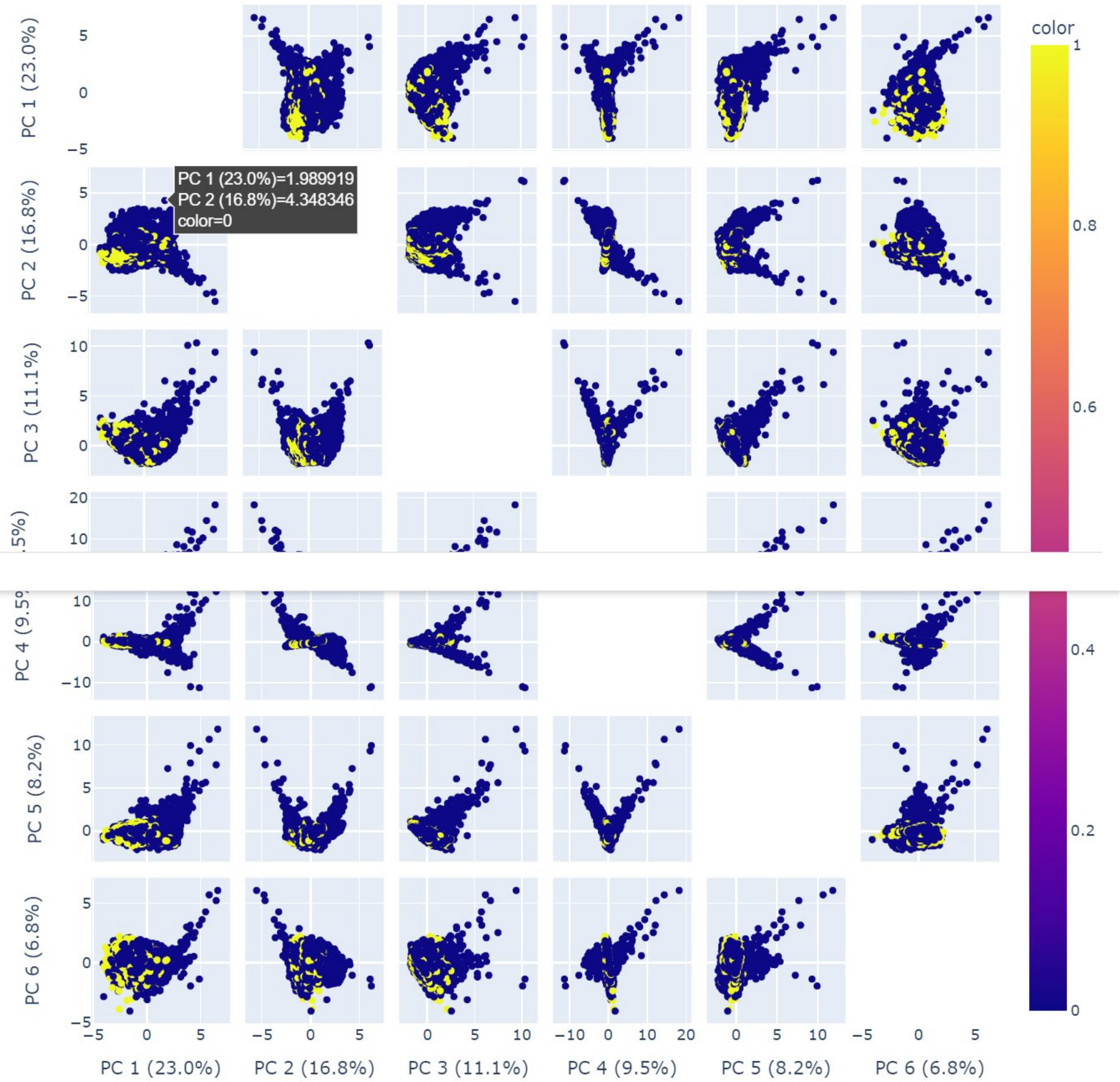


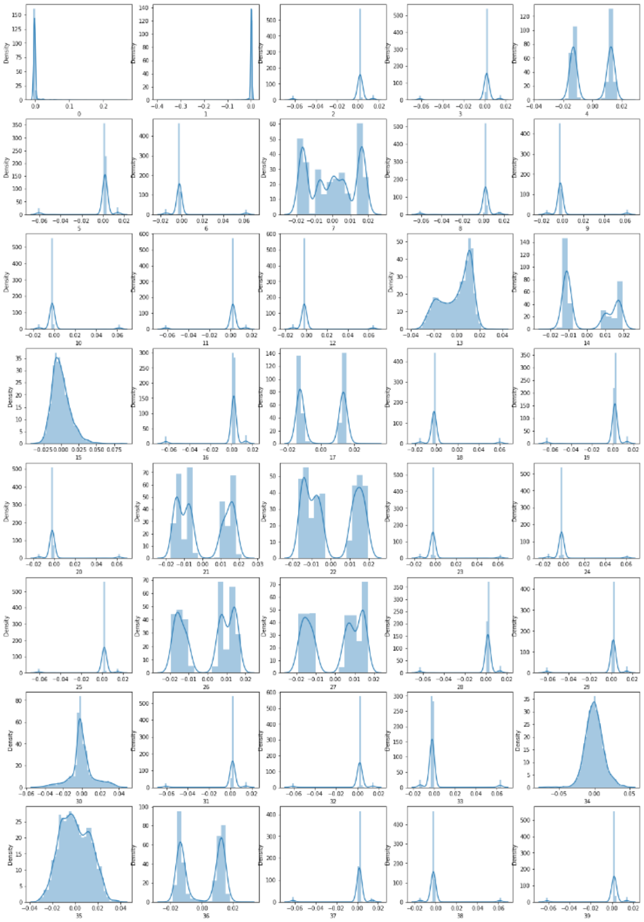
Fig 4.4 Distribution – Principal Components

3. INDEPENDENT COMPONENT ANALYSIS

Independent component analysis (ICA) is a statistical and computational technique for revealing hidden factors that underlie sets of random variables, measurements, or signals.

ICA defines a generative model for the observed multivariate data, which is typically given as a large database of samples. 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 nongaussian 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.

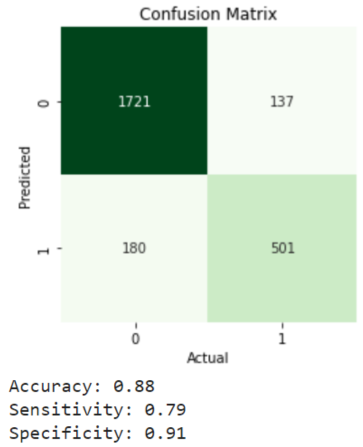
Fig 4.5 Distribution Plot - ICA



ICA is superficially related to principal component analysis and factor analysis. ICA is a much more powerful technique, however, capable of finding the underlying factors or sources when these classic methods fail completely. In ICA, the features are not ordered. Kurtosis is used to determine importance of features in ICA. When kurtosis is positive, the variable is said to be super-Gaussian or leptokurtic. Super-Gaussians are characterized by a spiky pdf with heavy tails, i.e., Laplace pdf. When kurtosis is negative, the variable is said to be sub-Gaussian or platykurtic. Sub-Gaussians are characterized by a rather flat pdf. Below chart Fig 4.5 show the distribution of all independent components. We can see that 27 features show pdfs with a spike.These features determine the features locally.Since the dataset is linearly separable globally, ICA does not perform well in classifying the outcome.

4. RANDOM PROJECTIONS

A Random Projection can be used to reduce the complexity and size of data, making the data easier to process and visualize. It is also a preprocessing technique for input preparation to a classifier or a regressor. Random Projection is typically applied to highly-dimensional data, where other techniques such as Principal Component Analysis (PCA) can't do the data justice. The method generates a new dataset by taking the projection of each data point along a randomly chosen set of directions. The projection of a single data point onto a vector is mathematically equivalent to taking the dot product of the point with the vector



```
Accuracy: 0.88
Sensitivity: 0.79
Specificity: 0.91
```

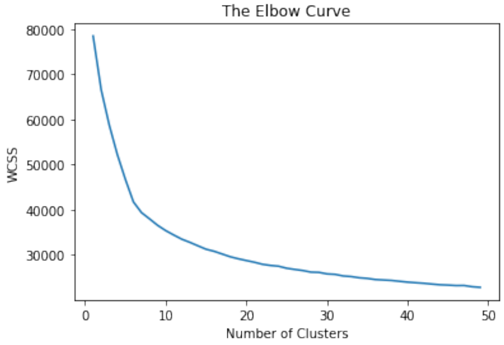
From Fig 4.6, the features selected is same as that of PCA(which is 13 features). The best parameters chosen for no of neurons of 25 with cross validation accuracy of 0.80. Test Accuracy is 0.88 which is an improved performance when compared to PCA.

Fig 4.6 CM – Random Projection

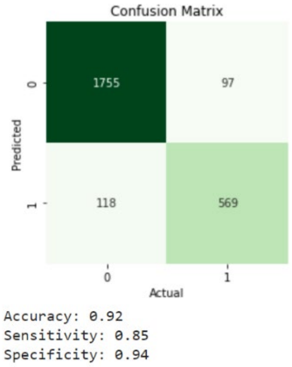
CLUSTER ANALYSIS USING REDUCTION TECHNIQUES

1. K-MEANS ON FEATURES SELECTED USING RANDOM FOREST

Fig 5.1 Elbow Method



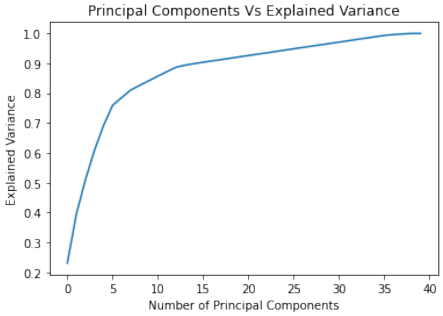
K-Means clustering is done on the features selected using Random Forest. The features are selected in Random Forest based on feature importance. Using elbow method in Fig 5.1, we can see the WCSS with respect to number of clusters and we choose the optimum value as 10. Dummy variables are created with respect to the cluster member and added in addition to the features chosen in the Random Forest.



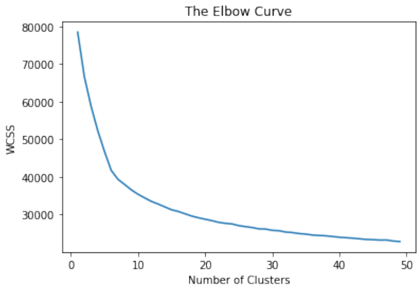
```
Accuracy: 0.92
Sensitivity: 0.85
Specificity: 0.94
```

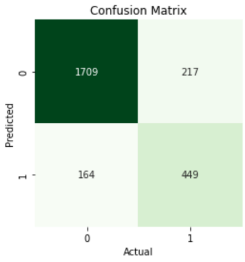
Best chosen parameter for number of neurons is 30 and cross validation accuracy is 0.85. The test accuracy showed is 0.92 which showed an improvement in accuracy with respect to model of features selected using RF.

2. K-MEANS ON PCA



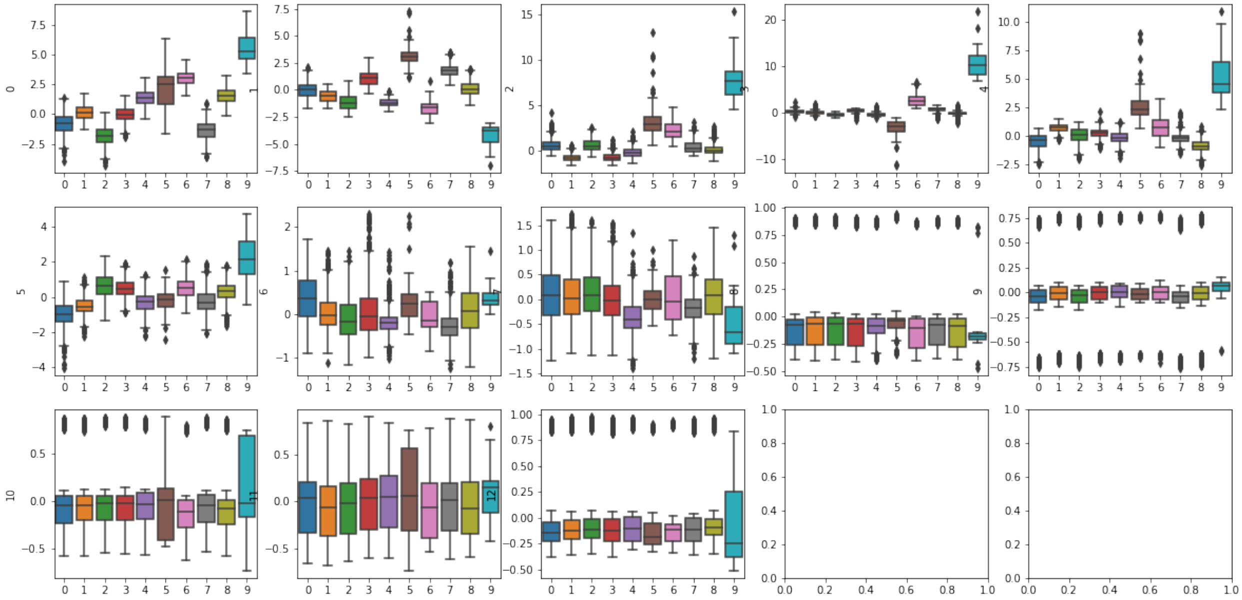
K-Means is performed on the dataset where features have been chosen based on Principal Component Analysis. 13 features were selected for the analysis.Using elbow method, we have chosen the optimum number of clusters to be 10. Dummy variables are created based on the cluster members.From the below box plot Fig 5.2, we can see that the clusters are circular. Best chosen parameters for no of neurons are 20 with cross validation accuracy as 0.77. The model shows a better test accuracy of 0.86 which shows an improvement with respect to PCA features.





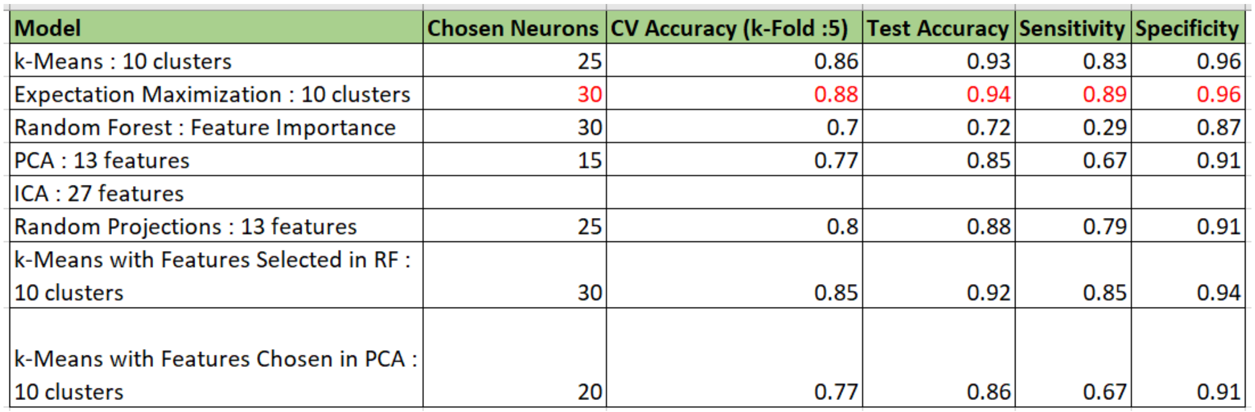
```
Accuracy: 0.86
Sensitivity: 0.67
Specificity: 0.91
```

Fig 5.2 Box Plot Distribution – kMeans with PCA



COMPARISON OF CLASSIFIERS

From below Fig 6.1, we can see that the model with Expectation Maximization have given a good predictability which shows the highest accuracy among other models.



Model	Chosen Neurons	CV Accuracy (k-Fold :5)	Test Accuracy	Sensitivity	Specificity
k-Means : 10 clusters	25	0.86	0.93	0.83	0.96
Expectation Maximization : 10 clusters	30	0.88	0.94	0.89	0.96
Random Forest : Feature Importance	30	0.7	0.72	0.29	0.87
PCA : 13 features	15	0.77	0.85	0.67	0.91
ICA : 27 features					
Random Projections : 13 features	25	0.8	0.88	0.79	0.91
k-Means with Features Selected in RF : 10 clusters	30	0.85	0.92	0.85	0.94
k-Means with Features Chosen in PCA : 10 clusters	20	0.77	0.86	0.67	0.91

Fig 6.1 Comparison of Classifiers

The bike sharing dataset is linearly separable, hence most of the models with original features perform similarly. Expectation maximization model performs the best in terms of overall metrics. The performance of k-Means clustering with PCA has shown a better result with respect to k-Means with original features. Also, Random Component Analysis (Random Projections) performed better when compared to PCA and ICA dimension reduction techniques. Random projection are also quite fast for reducing the dimension of a mixture of Gaussians. If the data is very large, you don't need to hold it in memory for a random projections, whereas for PCA you do. In general PCA works well on relatively low dimensional data. PCA gives the best projection for some initial training data but it might become arbitrarily worse as time goes by and new data arrives with an "evolved" distribution. Random projections give a kind of probabilistic warranty against that situation. But, cluster number k might become too low if d is increasing over time but for continuously learning from large streams of data, random projections are a sensible and efficient approach.

CONCLUSION

For the binary classification problem, model has been generalised well based on the algorithms that has been introduced, k-Means Clustering and ANN.For ANN, we initialized the ANN by calling the *Sequential* class from the *Keras* library in Python. Keras comes in combination with Tensorflow library. Then. We have added the fully connected Input Layer to the Sequential ANN by calling Dense class from Keras. We also fully connected Output Layer in the same way.For the first layer, Rectifier Activation function is used followed by Sigmoid Function in the output layer to find the probability.

For K-Means Clustering, we first initially found the value of optimum value of k for the dataset.*Elbow Method* is used for this. This method plots the value of inertia produced for different values of k. As k increases, the value of inertia decreases. The criteria to choose the value of k using the graph is find the value of k after which the inertia does not decrease significantly anymore. Other known methods are Silhouette Coefficient and Davies-Bouldin Index (DB Index) etc.

For our classification problem, for each clusters formed for different models, the total no of predictions (true/false) per cluster has been noted. From the above analysis, we can see that the clusters formed are round and *compact*. The total *within-cluster sum of square* measures the compactness (i.e goodness) of the clustering and we want it to be as small as possible. We could see that clusters created with feature importance(using RF) and PCA were more round and compact (WCSS as 13380.28 and 26655.37) than normal k-Means Clustering(WCSS:35375.02). The clusters also showed which data points are in positive class and have impact on certain features. For example, cluster 2 has high temperature, radiation , low humidity in normal K-Means Clustering and for EM, Cluster 0 and 5 show high distribution in snowfall and rainfall which makes this feature not suitable to rent bikes.

Feature Selection in ICA is purely done on checking the *variance* of the features. Feature selection in RP is done using *Gaussian* or Sparse techniques. We have used Gaussian Projection that reduces the dimensions of high-dimensional data by projecting the original input's dimensional space onto a randomly generated matrix. Though 27 features were selected in ICA, the performance with ICA was not good enough. Random Projection, however, showed a better test accuracy than PCA with a value of **0.88.** RCA showed faster execution than PCA model.

The transformations on features did not improve the accuracy of the model. The feature transformation done for the purpose of dimensionality reduction has given a poor performance Techniques used were PCA, ICA and Random Projection. In PCA, 13 features were selected based on the *principal components* and its corresponding Eigen values. However, selecting all the features in PCA came to give same results as that of models with original features.With 13 features, the test accuracy was **0.85.**

On comparing all models, *Expected Maximisation* performance is better than all the other algorithms.This gave a test accuracy of 0.94. Several factors made this algorithm better than the rest. One of them is it naturally produces valid parameters for the mixture distribution on every iteration. In contrast, other optimization algorithms would need constraints to be imposed. K-Means algorithm with dimension reduction technique (PCA) did show a better improvement than without dimension technique.

Agglomerative Hierarchical Clustering and *k-Mediods* can be conducted to see if there is any improvement in the performance of the clustering based algorithms. Metrics like *Homogenuity score, Completeness score, V measure score* can be validated for giving cluster labelling the ground truth of homogenuity and completeness.