Aykut Aykut
0034039
January 9th, 2019
Comp421 Homework07

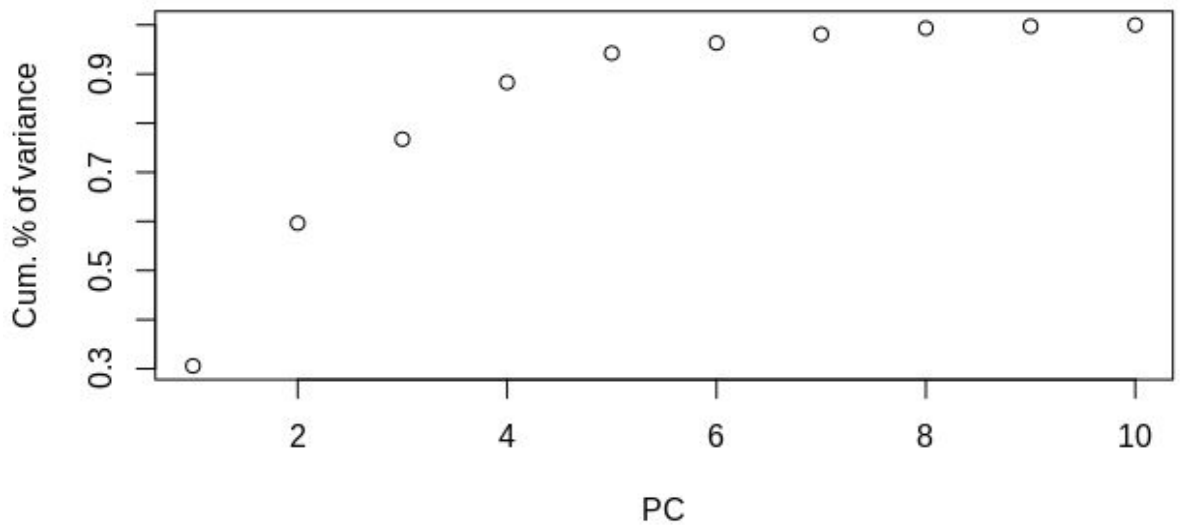
In this homework, I was asked to implement a machine learning solution for the given problem. In order to do that, I followed these 10 steps:

1) I loaded required libraries AUC, MASS and xgboost at first.

2) I read the data files into the memory as *train_x, train_y* and *test_x.*

3) I examined the *train_x* and realised that features are from categorical variables, continuous variables and binary variables. I splitted *train_x* into *train_x_cont* which contains only continuous features and *train_x_rest* which contains non-continuous features.

4) I applied PCA on continuous features with *center = TRUE* and *scale = TRUE,* because each feature has different min, max and mean values. I plotted principal components vs. cumulative percentage variance figure below, and realised that the first 6 PCs explain the 96% of variances from summary of PCs. Therefore, I reconstructed the continuous features with 6 PCs by using the formula below. After that, I column binded continuous features on which PCA is applied, and *train_x_rest.*



```
> summary(train_pca)
Importance of components:
                          PC1     PC2     PC3     PC4     PC5      PC6      PC7      PC8      PC9      PC10
Standard deviation     1.7485 1.7047 1.3075 1.0747 0.77235 0.45621 0.42129 0.35132 0.19833 0.16533
Proportion of Variance 0.3057 0.2906 0.1709 0.1155 0.05965 0.02081 0.01775 0.01234 0.00393 0.00273
Cumulative Proportion  0.3057 0.5963 0.7673 0.8828 0.94243 0.96324 0.98099 0.99333 0.99727 1.00000

train_x_cont_pca <- train_pca$x[, 1 : pca_dim] %*%
  t(train_pca$rotation[, 1 : pca_dim])
train_x_cont_pca <- t((t(train_x_cont_pca) * train_pca$scale) + train_pca$center)
```

5) I created a grid search table with the following parameters:
   a. *eta: [0.1, 0.4, 0.7, 1]*
   b. *gamma: [0, 5, 10, 20]*
   c. *max_depth: [4, 8, 16, 32]*
   d. *nrounds: [10, 20]*

6) I used *xgb.cv* function in my grid search algorithm. In grid search, I used 8-fold cross validation with stratification since the train data is unbalanced, auc as the performance metric and *early_stopping_rounds = 5*. According to grid search results, the following parameters performed best auc in both train and validation set:
   a. *eta: 0.4*
   b. gamma: 20
   c. max_depth: 8
   d. nrounds: 10

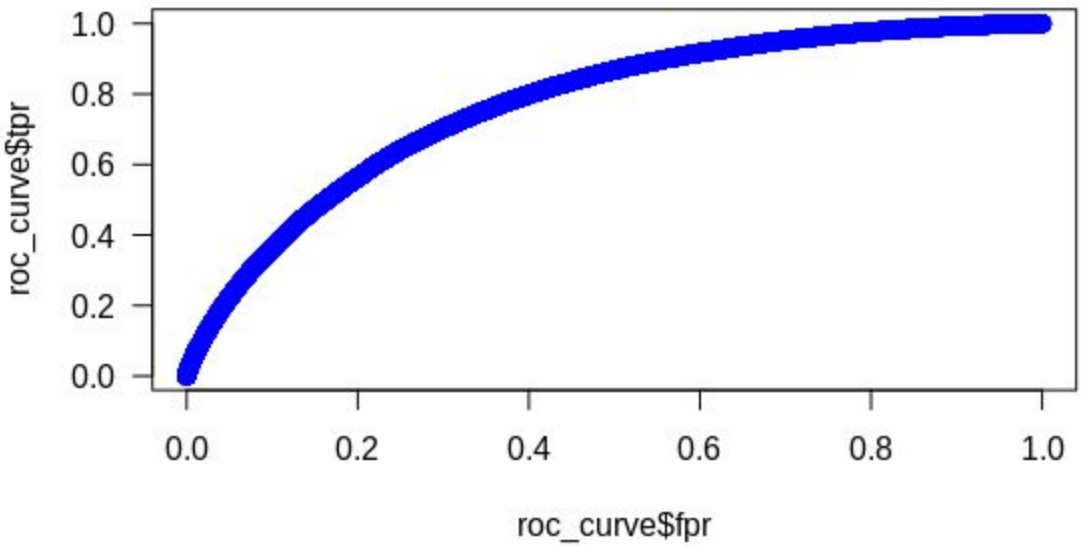
7) I fitted my xgboost model with the best parameters to the whole *train_x* and predicted probability of positive class.

8) I calculated the confusion matrix and printed it on the console.

```
> print(confusion_matrix)
                 train_y
xgb_predicted_label      0      1
                 0 253550  43927
                 1   1100   1423
```

9) I calculated AUROC, plotted it as follows and printed on the console.

```
> auc(roc_curve)
[1] 0.7683644
```



10) I applied the same preprocessing steps for the test data, feature examination and PCA. Then predicted positive class probabilities for the *test_x* and wrote them on file named *0034039_comp421_hw07_test_predicted.csv.*