HOMEWORK 7

Rohith Reddy Kolla | rkolla@buffalo.edu

1. The Cleveland data contains 296 observations and 15 variables. Since diag2 is disregarded, it is removed from the data. The data is then split into training and test sets in 2:1 ratio respectively.

- Neural Network:

  Since the Cleveland data has categorical data, the test and training sets are converted using the dummyVars function from the caret package so they can be used with the neural network.

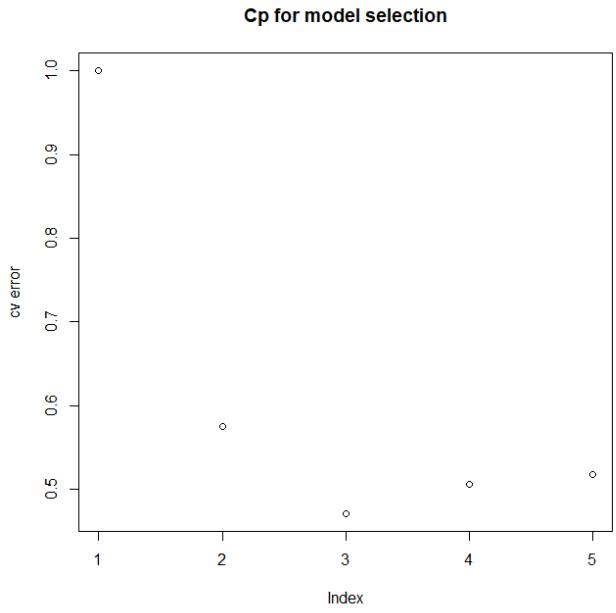
  Using the neuralnet package 5 neural networks with a single hidden layer are fitted on the training set with a stepmax of 10^9 and number of nodes in the hidden layer ranging from 1 to 4. The resulting training errors and test errors are shown below.

```
> train_err_store
[1] 0.47979798 0.09090909 0.07575758 0.03535354 0.20202020
> test_err_store
[1] 0.4183673 0.2142857 0.1530612 0.2142857 0.2857143
```

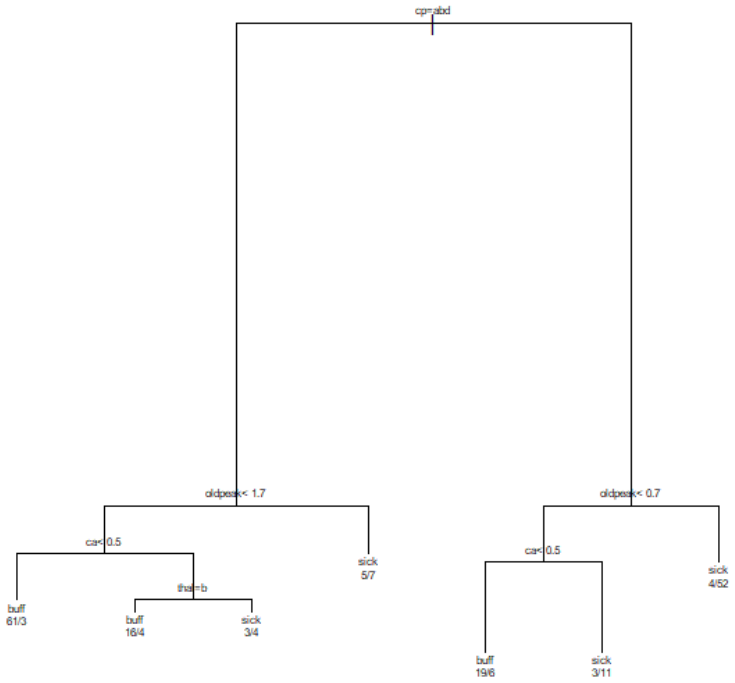
  From the errors shown above, it can be seen that the training error is decreasing with an increase in the number of hidden nodes with a significant decrease between the first and second neural net. Furthermore, it is evident that the third neural network containing 3 hidden nodes performs the best in predicting the test set with an error of around 15.3%. It is important to note that different training and test splits could provide significantly different errors. Also, we cannot gather an intuitive understanding of how important the neural network considers each of the predictors to be.

- CART:

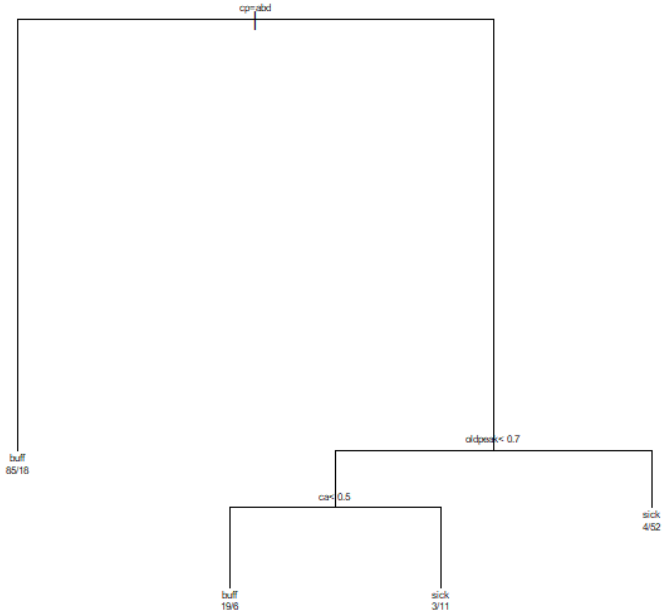
  For the CART model, a tree is fit on the training data using rpart's default minsplit and cp = 0. The tree is then pruned based on the cp value providing the least cv error. The structure of the two trees along with the plot indicating the cv error for the different cp values are shown below.



Initial Tree -



Pruned Tree –

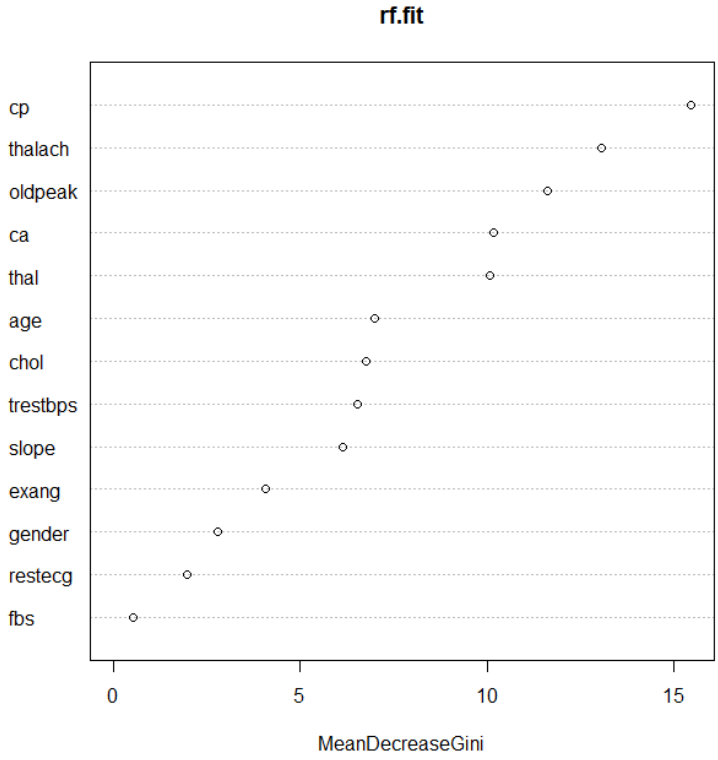


The resulting error rate for the test set using the pruned tree comes out to be 26.5% which is considerably worse than the results from using the neural network with 3 hidden nodes. However, unlike the neural network, we can tell what variables the cart model considers to be important. The variable importance is shown below and we can see that the model considers the cp variable to be the most important followed by thelach and oldpeak whereas age, chol and fbs are the least important.

```
Variable importance
     cp thalach oldpeak    thal   slope   exang      ca     age    chol     fbs
     26      16      16      12      11      11       6       1       1       1
```

- Random Forest:

Lastly, a random forest model is fit on the training data with the number of trees = 5000. The resulting error rate for the test set is 19.3% which is better than the cart model but still worse than the neural network. The plot showing the variable importance for the random forest model is shown below. Similar to the cart model, the most important variables turn out to be cp followed by thalac and oldpeak. However, age and chol are considered to be relatively more important with the least important variables being gender, restecg and fbs.



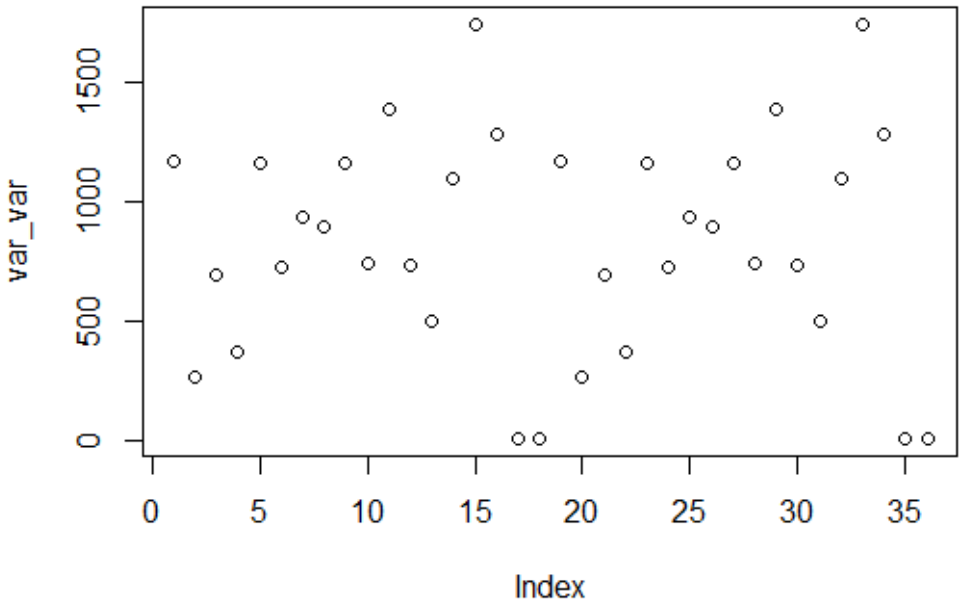
	MeanDecreaseGini
age	6.9974009
gender	2.8125218
cp	15.4674326
trestbps	6.5349132
chol	6.7580171
fbs	0.5368866
restecg	1.9835233
thalach	13.0539767
exang	4.0919294
oldpeak	11.6065532
slope	6.1288477
ca	10.1782060
thal	10.0737641

2. a)

The pendigits dataset contains 10992 observations and 36 variables with the last variable indicating the class label (digits). The variance of each the 35 variables excluding the class label variable is shown below.

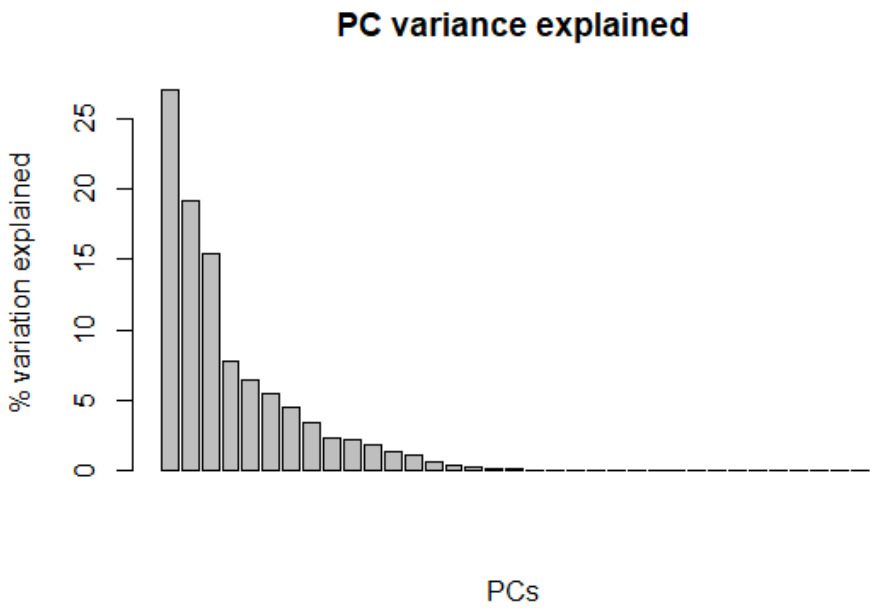
```
        v1          v2          v3          v4          v5          v6
1173.595672  263.042035  693.952813  367.245343 1162.845095  728.821144
        v7          v8          v9         v10         v11         v12
 934.945662  894.375033 1165.234377  742.646884 1390.401333  735.493418
       v13         v14         v15         v16         v17         v18
 498.876311 1099.284727 1743.930994 1280.071975    8.276822    8.408414
       v19         v20         v21         v22         v23         v24
1173.595672  263.042035  693.952813  367.245343 1162.845095  728.821144
       v25         v26         v27         v28         v29         v30
 934.945662  894.375033 1165.234377  742.646884 1390.401333  735.493418
       v31         v32         v33         v34         v35         v36
 498.876311 1099.284727 1743.930994 1280.071975    8.276822    8.408414
```

Plotting the above data in the plot below shows that the variance of many of the variables are similar to each other and most of them have a variance between 500 and 1500.



The data is scaled and the prcomp function is used to extract the principal components. The percentage of variance explained by each principal component is shown below.

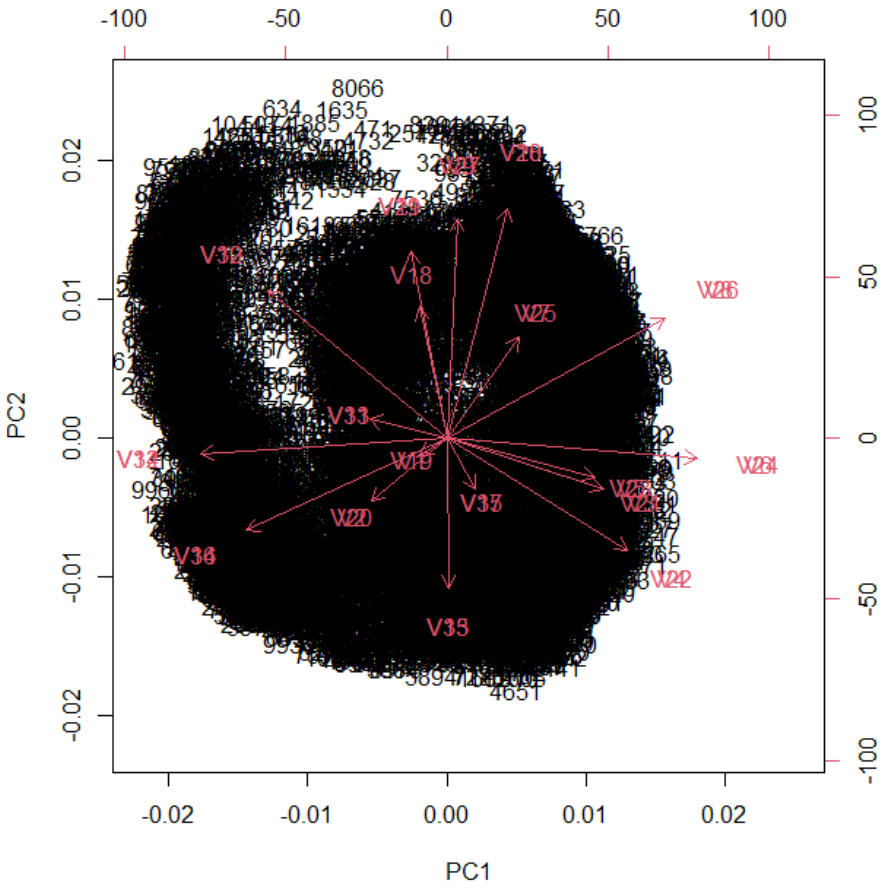
```
 [1] 2.702400e+01 1.920035e+01 1.543685e+01 7.787427e+00 6.435033e+00 5.469518e+00 4.526803e+00
 [8] 3.423792e+00 2.366926e+00 2.254916e+00 1.910951e+00 1.336732e+00 1.114266e+00 6.339007e-01
[15] 4.297869e-01 3.280289e-01 1.919040e-01 1.288180e-01 2.423109e-27 5.152657e-28 4.593143e-29
[22] 1.194066e-29 7.866725e-30 7.651652e-30 7.563808e-30 5.642634e-30 4.229408e-30 1.717951e-30
[29] 1.432336e-30 1.343129e-30 8.762568e-31 6.269945e-31 3.514063e-31 1.624285e-31 3.482726e-32
```



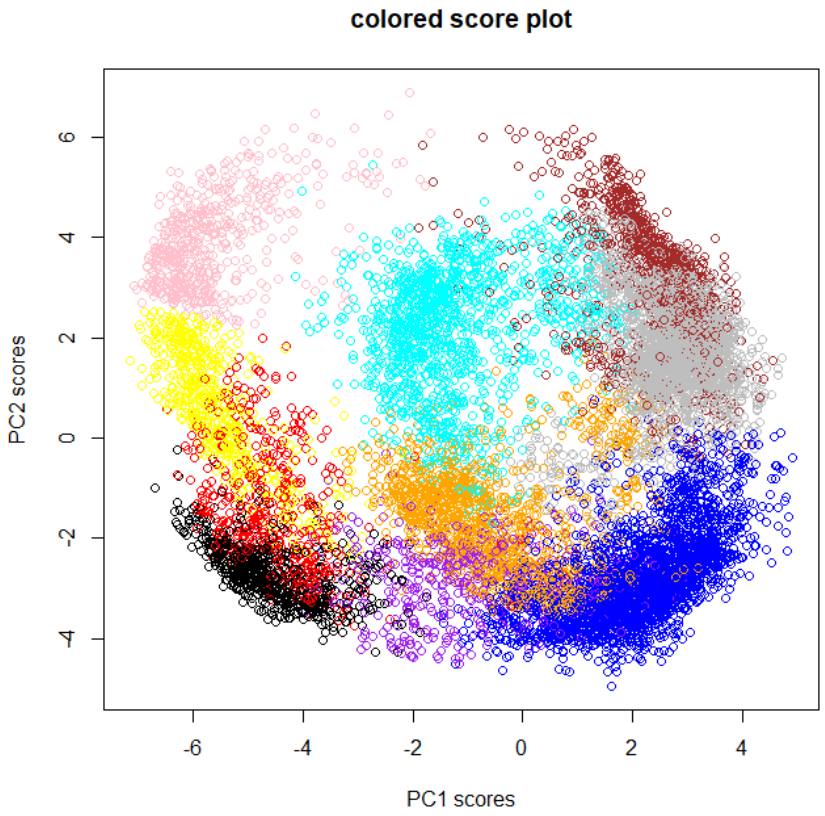
The number of PCs required to explain 80% of the variance is 6 and for 90% it is 9.

```
> sum(per_var_exp[1:9])
[1] 91.6707
> sum(per_var_exp[1:6])
[1] 81.35318
```

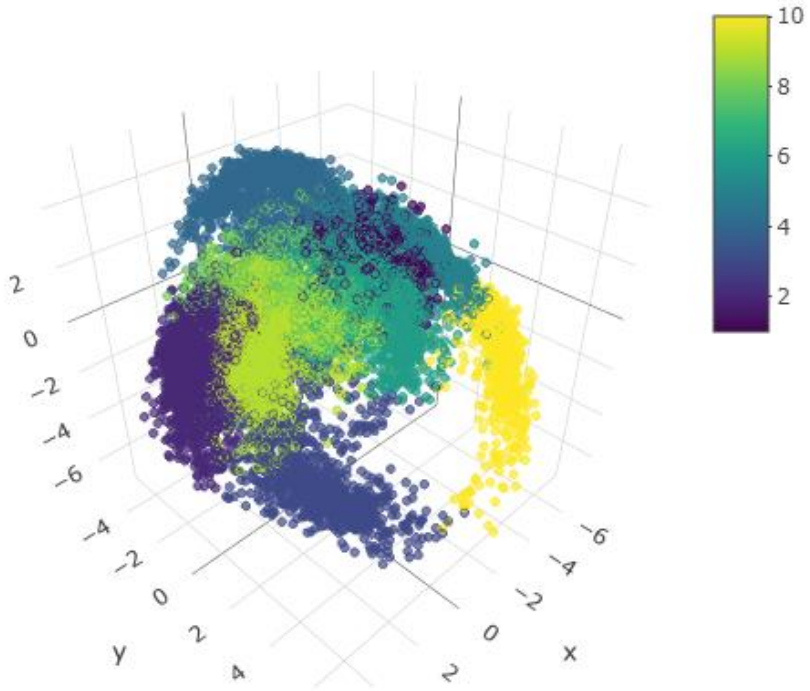
The biplot for the PCs is shown below. It is pretty unclear due to the large number of observations.



The two-dimensional score plot for PC1, PC2 is shown below with each color representing a different class.



The clear separation between each class is very evident. The separation is further shown in the 3d plot shown below.



b)

The raw data and the PCs that capture at least 80% of the variation are divided into test and training sets in 1:2 ratio respectively. The data is then fit into a KNN model using 15 as k and the prediction accuracy is shown below.

For the raw data - acc1 98.1168122270742

For the PC data with at least 80% variation - acc2 97.6255458515284

As we can see, KNN on the raw data works slightly better than the KNN on the PC data. However, the raw data with 35 columns is significantly larger than the PC data which has only 6 columns and the PC data captures only 80% variation. Considering this, the PC data works fairly well and goes to show that principal components can be applied to reduce the data involved significantly while maintaining a similar accuracy as raw data.

3)

a)

Cross validation is implemented by imitating new independent data set by leaving out points one at a time and estimating the solution based on the remaining points.

b)

In k-fold cross validation the data is split into k different parts and the model is fit on k-1 parts and the remaining part is used for the evaluation.

i) The hold out method being the simplest kind of cross validation has the advantage of being easy to implement and computationally inexpensive compared to k-fold. However, the results of the hold out method are more susceptible to poor splitting of the dataset and the evaluation has higher variance relatively.

ii) Since in LOOCV the model is fit into all observation except 1 and the remaining is used for evaluation, it is computationally more expensive in general than k-fold but it provides better results.