MATH 642 Project
Zijing Gao

Introduction:
Glass identification and classification is applied when searching for the evidence at the scene of the crime. The glass left can be used as the evidence if the glass type is correctly identified. Correct identification can be helpful to improve the efficiency for cracking a criminal in a lot of cases. If you are interested in this, please click here to check.

Problem Description:
The objective of this project is to identify and classify the type of the glass. There are several types of the glass which can be hard to identify by professional approaches since the glass in the criminal scene is fragment or covered by other materials. My goal is to use the amount of materials that consists of the fragments of glass, such as Fe, Al, Mg and so on to identify the type of glass. Since there are many kinds of glasses to take into account, this task can be challenging.

I assume the customer will be the criminological investigation since correct classification is greatly helpful in glass analysis. After correct identification of glass type, it can be helpful when determining direction and sequence of force.

This type of determination compares a known sample to a glass fragment to see if the two samples came from the same source. Glass can be made from a variety of different materials that differ from batch to batch. The presence of the different materials in the glass makes it easier to distinguish one sample from another. Also, the properties of glass can vary depending upon the temperature the glass is exposed to during manufacturing. Basic properties, such as color, thickness, and curvature, can also help to identify different samples of glass just by looking at them. Optical properties, such as refractive index (RI), are defined by various manufacturing methods. RI is the manner in which light passes through the glass. This can be measured easily even on small fragments of glass. These properties help to indicate that two samples of glass could be from the same source.

Data Description:
- Title : Glass Identification Database
- Sources:

Here is the data source

(a) Creator: B. German
-- Central Research Establishment
Home Office Forensic Science Service
Aldermaston, Reading, Berkshire RG7 4PN
(b) Donor: Vina Spiehler, Ph.D., DABFT
Diagnostic Products Corporation
(213) 776-0180 (ext 3014)
(c) Date: September, 1987

There is no missing/ambiguous data in this dataset.

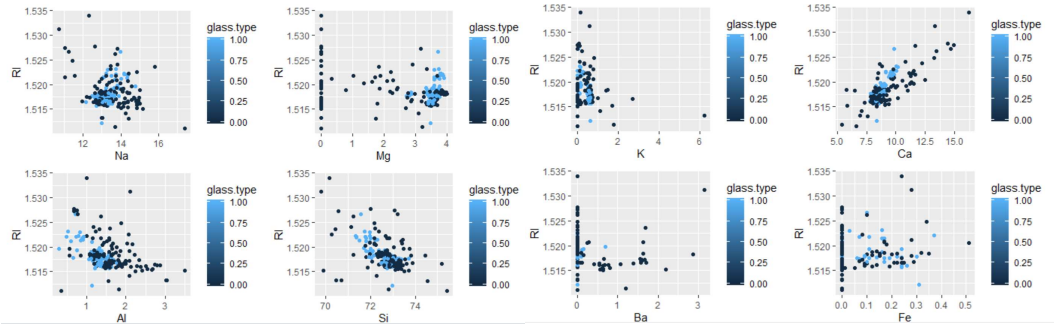
- Challenges:
  There are 7 types of glasses for classification
  Type of glass: (class attribute)
  -- 1 building-windows float processed
  -- 2 building-windows non-float processed
  -- 3 vehicle-windows float processed
  -- 4 vehicle-windows non-float processed (none in this database)
  -- 5 containers
  -- 6 tableware
  -- 7 headlamps

  And we have attribute information
  Attribute Information:
  1. Id number: 1 to 214
  2. RI: refractive index
  3. Na: Sodium (unit measurement: weight percent in corresponding oxide, as are attributes 4-10)
  4. Mg: Magnesium
  5. Al: Aluminum
  6. Si: Silicon
  7. K: Potassium
  8. Ca: Calcium
  9. Ba: Barium
  10. Fe: Iron

  It is hard to reduce the accuracy of the classification error since there are 7 types of glasses. Model selection and error testing are both challenging since I need to test the validity and accuracy of the model and select the best model by comparison.

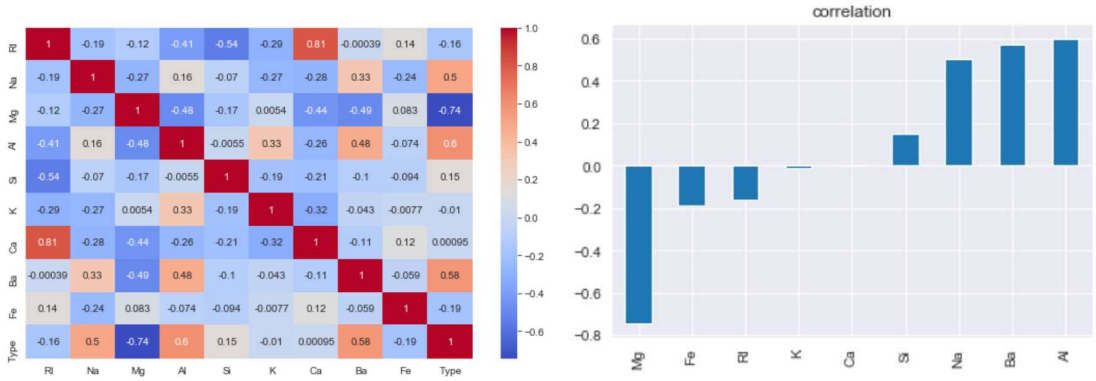
EDA:

At first, I try to simplify the target to classify the float and non-float glass. So I classify the Type 1 and Type 3 as “float” and others as “non-float” so that I can apply the model to a binary classification problem. Then, I plot the response (refractive index) with respect to each material in the glass because I think refractive index can be a good indicator of classification.



From the figures above, I find that the refractive index is linear related to Ca and Si.

Also, I created a heatmap and a bar plot to measure the correlation among these features.

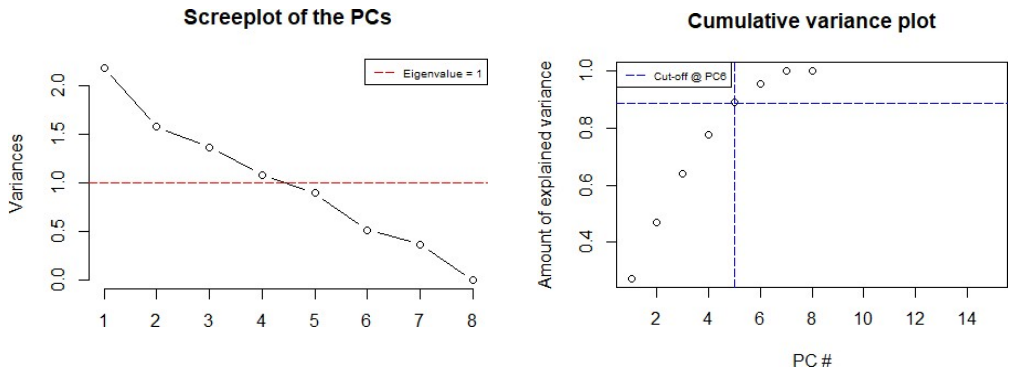


Here, I find that our target is related to Mg, Na, Ba and Al. I assume that the glass type is affected by these 4 materials much more severely than others.

Binary Classification results:

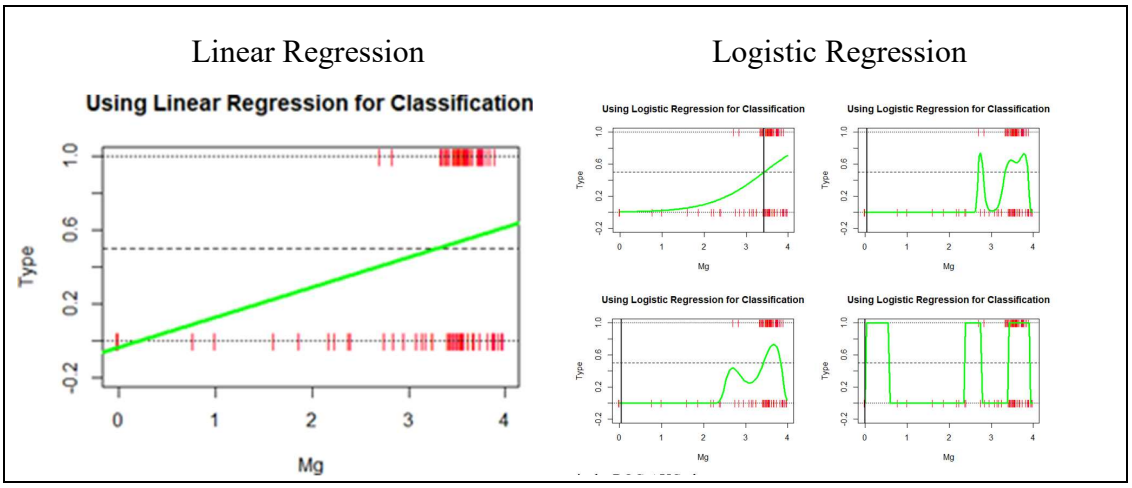
Data Preprocessing:

Then, I apply PCA on the attributes, hoping to reduce the feature columns to reduce the dimension. Here is the PVE figure.

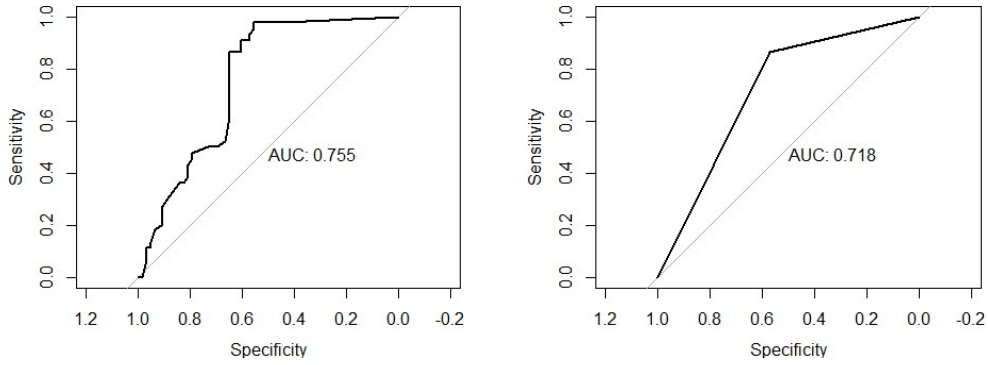


I find that the first 4 pricinpal components out of 8 can explain 80% variance so we can reduce the dimension to 4.

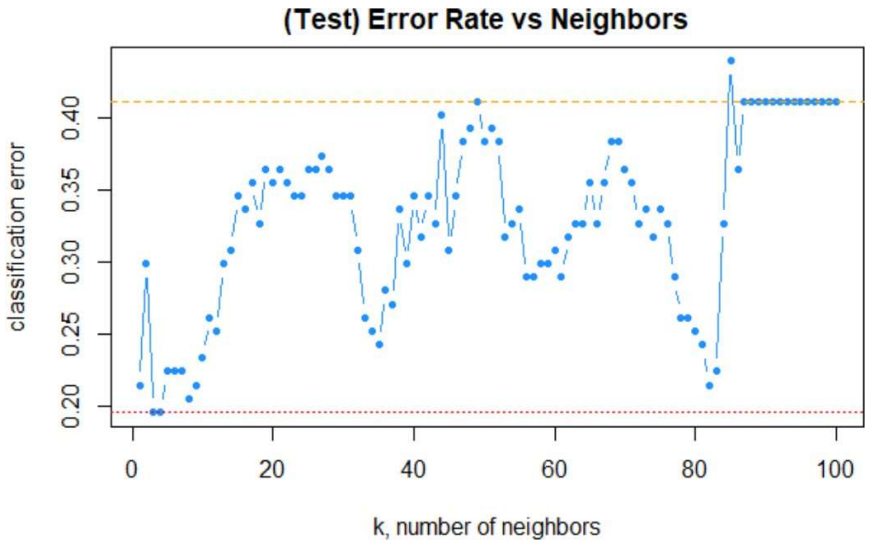
Supervised Learning:



At first, he linear model is not good since there are cases that the probabilities are below 0. Then, I apply LOOCV and K-fold cross validation on the train set using glm function and test the error for each polynomial. And I find the polynomial to be 4 which gives me the lowest cross validation (cv) error = 0.1788766. Also, I find the polynomial to be 4 or 9 when cv error = is 0.1788766 which is the lowest. The I calculate the MSE for each model and derive the relative plot with different polynomial.



At last, I apply K-Nearest Neighbors on the train set. After scaling and set k = 3 for a trial, I calculate the classification error which is 0.2336449, Then, I try to choose the best k and plot error vs choice of k.



Here, we choose k = 4 since the largest one is the least varibable and has the least chance of overfitting.

Then I apply SVM with different kernels and tune the parameters of the gamma and cost using tune function. Then I compute the confusion matrix for each case to select the best kernel which gives me the lowest classification error.

Also, I construct the Decision Tree and Random Forest algorithm along with bagging and boosting. After parameter tuning, the classification error is even lower than SVM with 10-fold cross validation.

At last, I construct an ANN with multiple choices of the number of the hidden layers. To avoid overfitting, I adjusted the number of layers and add some dropout layers and set the early stopping if needed. However, the shortage of observations results in the relatively unsatisfying classification result.

I plot the ROC curve for all of the classifiers I have applied for model selection.

Then, I extend my problem into a multiple classification problem which contains 6 glass types. The most challenging part is that the number of observations is not adequate to allow the model for learning thoroughly the relationship among the features and the response. However, I still apply all of the techniques if possible.

Analysis plan:

I simplify this multiple classification into binary classification in order to identify if the glass is floated. I try to use supervised learning and unsupervised learning to model the glass type distribution to identify glass type more accurately.

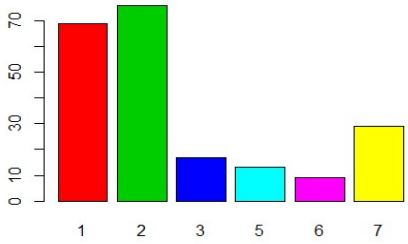
By comparing the sensitivity, recall, precision and classification error, I will select the one with the best performance on the validation set.

Also, I will apply almost the same techniques for the multiple classification problems and see what will happen.

Results

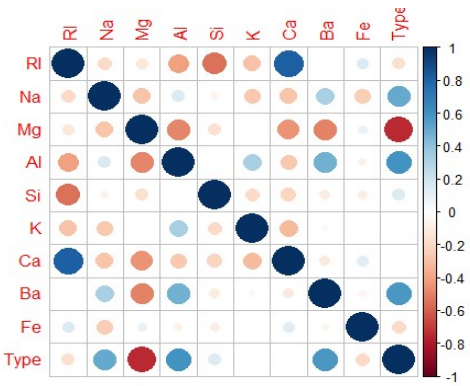
- Exploratory Data Analysis

Countplot



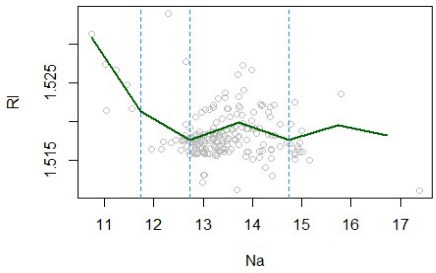
6 types of glass

Corplot

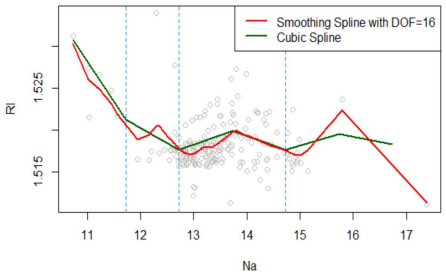


Glass type is much related to Na, Mg, Al and Ba

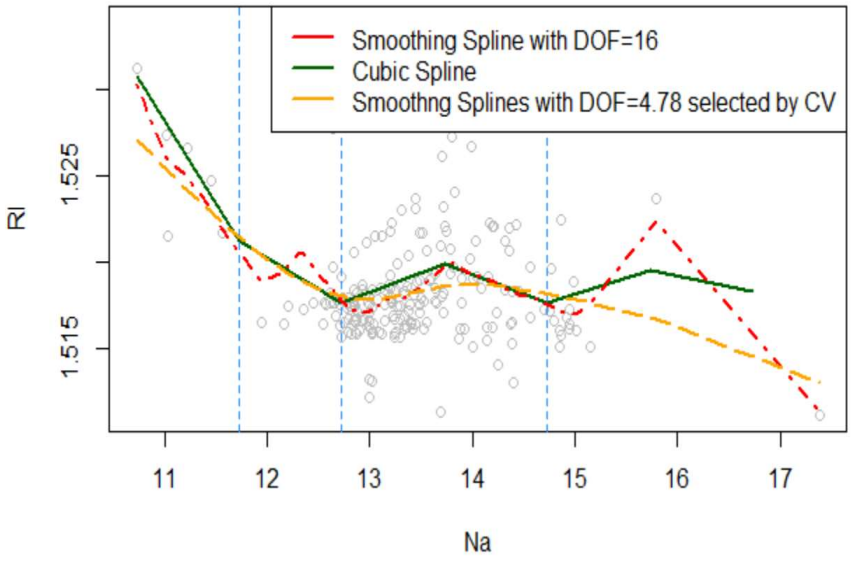
Spline



Smoothing Spine with DOF = 16

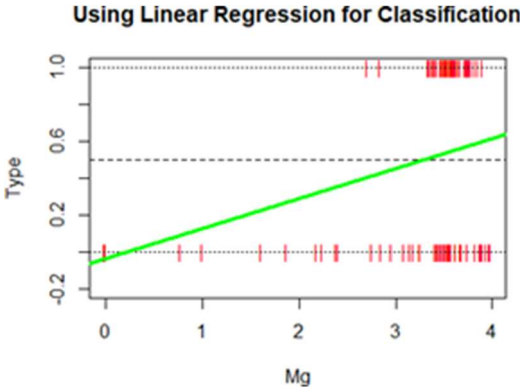


Smoothing Spline with DOF = 4.78 selected by CV



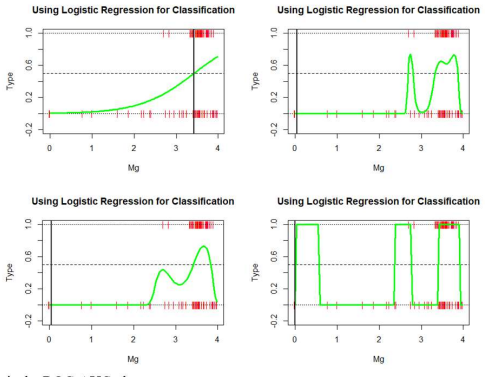
- Methods only for binary classification

Linear Regression



There are probabilities that are below 0

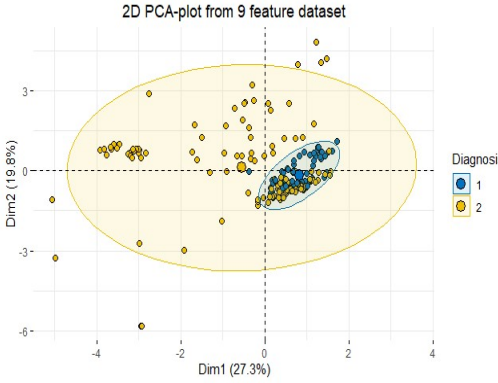
Logistic Regression



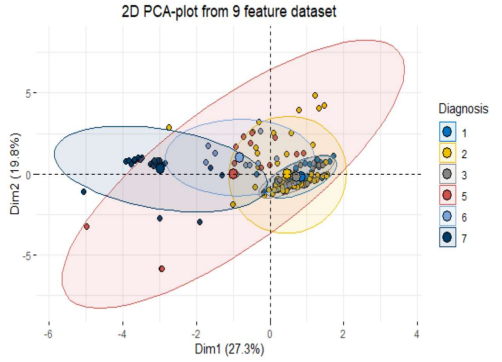
Sigmoid function is only for binary classification

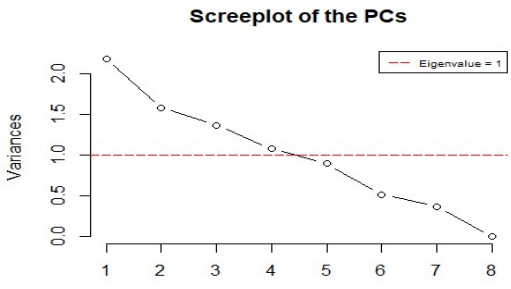
- Unsupervised learning results

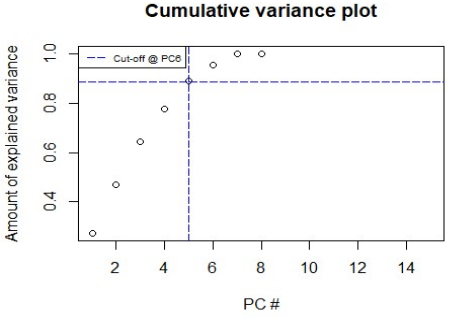
Binary



multiclass





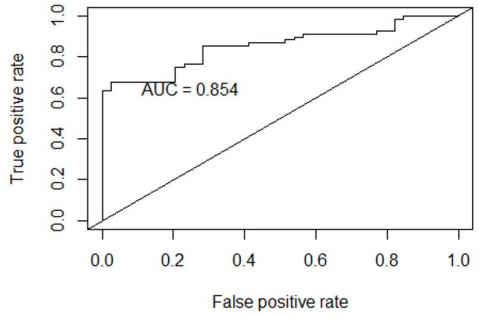


Here, I plot the 2D PCA plot from 9 features after searching for 4 principal components which explain almost 80% of the variance. The target is the rotate the data in a new axis to have a better visualization of the classification.
Based on the figures below, I transformed the original features into orthogonal base to explain the most variance and gain more information.

- Supervised learning results

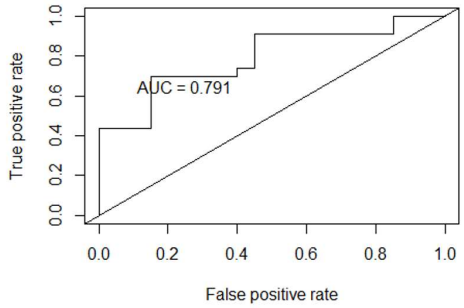
Binary	multiclass

1. LDA




```
          Reference
Prediction  1  2  3  5  6  7
         1 14  3  0  0  0  0
         2  4 11  1  3  0  0
         3  3  1  0  0  0  0
         5  0  2  0  1  0  0
         6  0  1  0  0  1  0
         7  0  0  0  0  0  7

Overall Statistics

               Accuracy : 0.6538
                 95% CI : (0.5091, 0.7803)
    No Information Rate : 0.4038
    P-Value [Acc > NIR] : 0.0002372

                  Kappa : 0.517
```

2. QDA




```
          Reference
Prediction  1  2  3  5  6  7
         1 15  1  1  0  0  0
         2 12  2  1  3  0  1
         3  3  1  0  0  0  0
         5  0  0  0  2  0  1
         6  0  1  0  0  0  1
         7  0  0  0  0  1  6

Overall Statistics

               Accuracy : 0.4808
                 95% CI : (0.3401, 0.6237)
    No Information Rate : 0.5769
    P-Value [Acc > NIR] : 0.9379

                  Kappa : 0.3018
```

3. SVM

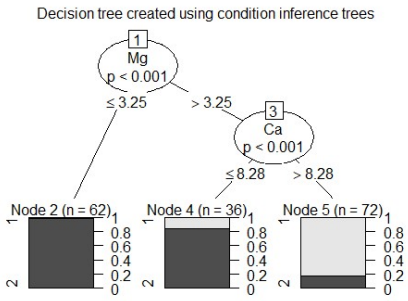
```
##           Reference
## Prediction  1  2
##          1 14  6
##          2  2 21
##
##                Accuracy : 0.814
##                  95% CI : (0.666, 0.9161)
##     No Information Rate : 0.6279
##     P-Value [Acc > NIR] : 0.006902
##
##                   Kappa : 0.6211
##
##  Mcnemar's Test P-Value : 0.288844
##
##             Sensitivity : 0.8750
##             Specificity : 0.7778
##          Pos Pred Value : 0.7000
##          Neg Pred Value : 0.9130
##              Prevalence : 0.3721
##          Detection Rate : 0.3256
##    Detection Prevalence : 0.4651
##       Balanced Accuracy : 0.8264
##
##        'Positive' Class : 1
##
## kernel:  radial
## accuracy:  0.8139535
## AUC:  0.8065217
```

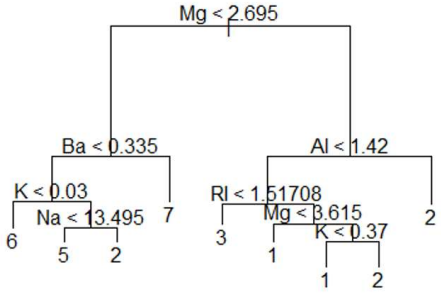
```
$accuracy
[1] 0.7692308

$table
          Reference
Prediction  1  2  3  5  6  7
         1 14  3  0  0  0  0
         2  3 16  0  0  0  0
         3  2  2  0  0  0  0
         5  0  1  0  2  0  0
         6  0  0  0  0  2  0
         7  0  1  0  0  0  6
```

```
## kernel:  radial
## accuracy:  0.7115385
## AUC:  0.9109932
```

Binary	Multiclass

4. Decision Tree





5. Random Forest

```
## No. of variables tried at each split: 9
##
## OOB estimate of  error rate: 10.59%
## Confusion matrix:
##    1  2 class.error
## 1 58  8 0.12121212
## 2 10 94 0.09615385
```

Multiclass After bagging by tuning mtry

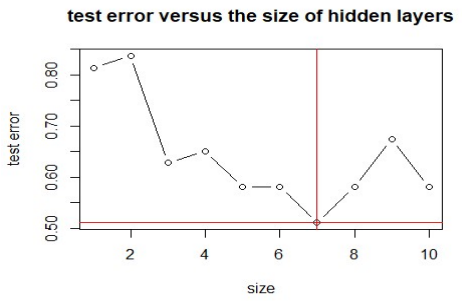
```
## OOB estimate of  error rate: 24.22%
## Confusion matrix:
##    1  2 3 5 6  7 class.error
## 1 41  9 1 0 0  1   0.2115385
## 2  7 42 3 3 2  0   0.2631579
## 3  4  2 7 0 0  0   0.4615385
## 5  0  0 0 9 0  1   0.1000000
## 6  1  2 0 0 4  0   0.4285714
## 7  1  2 0 0 0 19   0.1363636
```

Multiclass After boosting with n.trees = 5000

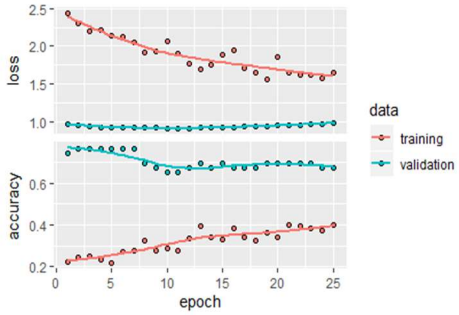
```
## yhat.bag  1  2  3  5  6  7
##        1 16  1  3  0  0  0
##        2  1 16  1  2  0  0
##        3  0  1  0  0  0  0
##        5  0  0  0  1  0  0
##        6  0  0  0  0  2  0
##        7  0  1  0  0  0  7

## [1] 0.8076923
```

```
          Reference
Prediction  1  2  3  5  6  7
         1 15  1  3  0  0  0
         2  2 17  1  2  0  0
         3  0  1  0  0  0  0
         5  0  0  0  1  0  0
         6  0  0  0  0  2  0
         7  0  0  0  0  0  7

Overall Statistics

               Accuracy : 0.8077
                 95% CI : (0.6747, 0.9037)
    No Information Rate : 0.3654
    P-Value [Acc > NIR] : 8.338e-11

                  Kappa : 0.7267
```

6. ANN

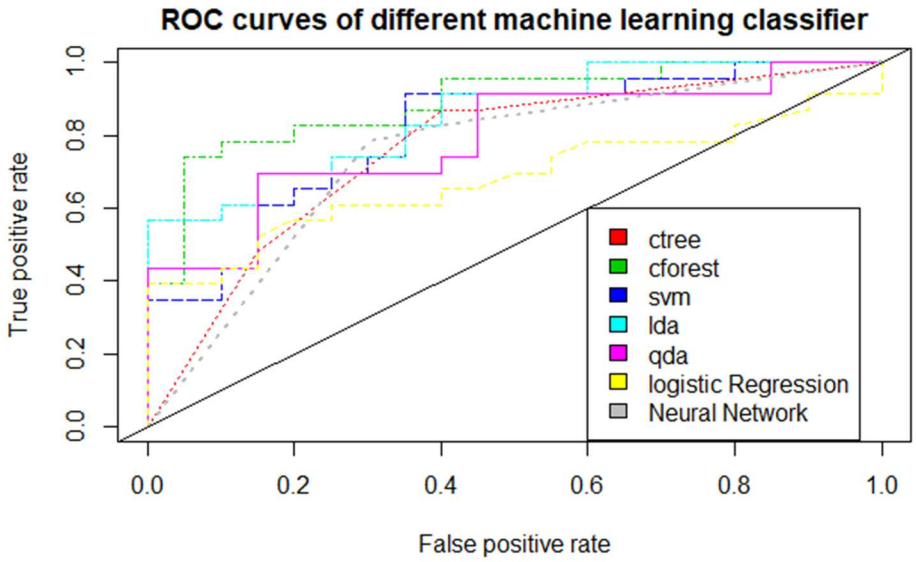


nnet using 10-fold cv (size = 7)



ANN using Keras with dropout layers

ROC



Based on the ROC curves, I find that the random forest method fits the best among these 7 methods, while Logistic Regression perform the worst.