wheat seed Analysis

Mohit Mehndiratta

10/16/2021

Loading the necessary libraries

library(boot)  
library(tree)

### Loading the data set and setting the seed value.

wheatKernelDS <- read.csv("wheat\_kernels.csv")  
wheat\_modified <- wheatKernelDS ## created a copy of the data set for later use  
attach(wheatKernelDS)  
set.seed(39)

## Data Exploration

### understanding the data set and exploring the variables.

head(wheatKernelDS)  
dim(wheatKernelDS)  
summary(wheatKernelDS)  
wheatKernelDS$Type <- as.factor(Type)  
str(wheatKernelDS)

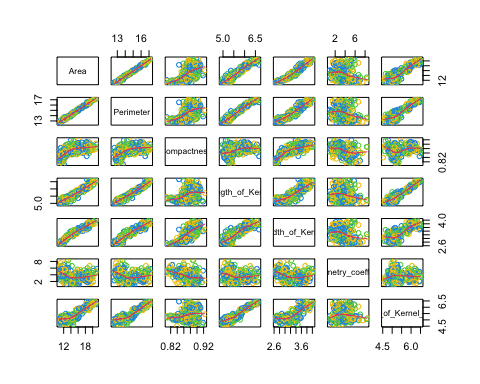
In this data set, there are 210 observations of 8 variables. The variables are the geometrical properties of wheat seeds. Except Type all the variables are numerical in nature. Type is a factor variable of 3 levels. Type 1 represents “Kama” wheat seeds, Type 2 represents “Rosa” wheat seeds and Type 3 represents “Canadian” wheat seeds.

### check if there is any null value.

## checking if there is any null value in the data set  
sum(is.na(wheatKernelDS))

### explore the matrices of the variables with plot

## see the pairs  
pairs(wheatKernelDS[,1:(length(wheatKernelDS) - 1)],   
 panel = panel.smooth, col = c(3,4,7))



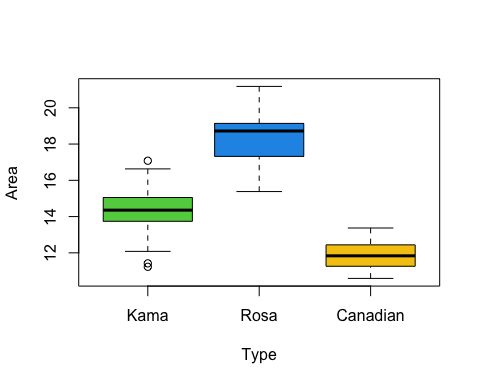
### understanding the relationship and variance within the variables

## Correlation matrix  
cor(wheatKernelDS[,1:(length(wheatKernelDS) - 1)])  
## Covairance matrix  
cov(wheatKernelDS[,1:(length(wheatKernelDS) - 1)])

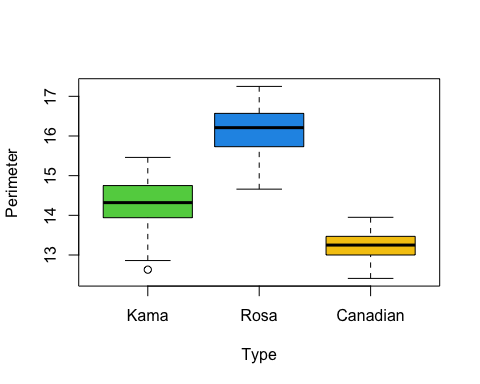
we can observe that each variables are positively correlated except the Assymetry Coefficient which is negatively correlated with each other.

### creating the boxplot to visualise each variables with each Type

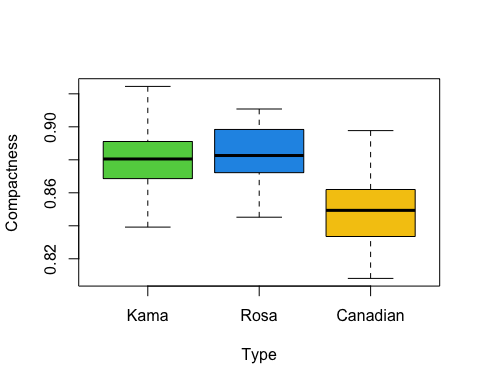
## Creating a box plot visualisation  
labels <- c("Kama", "Rosa", "Canadian")  
boxplot(Area~Type,names = labels, col = c(3,4,7))



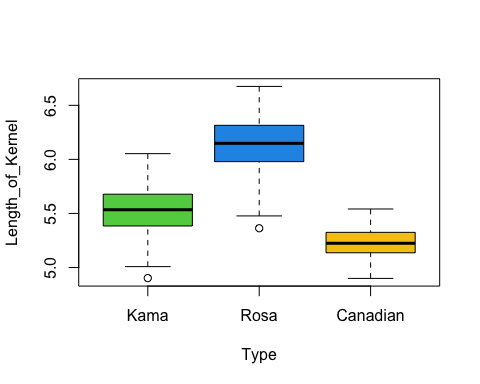
boxplot(Perimeter~Type,names = labels, col = c(3,4,7))



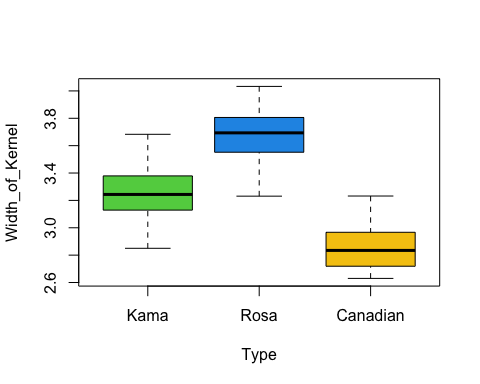
boxplot(Compactness~Type,names = labels, col = c(3,4,7))



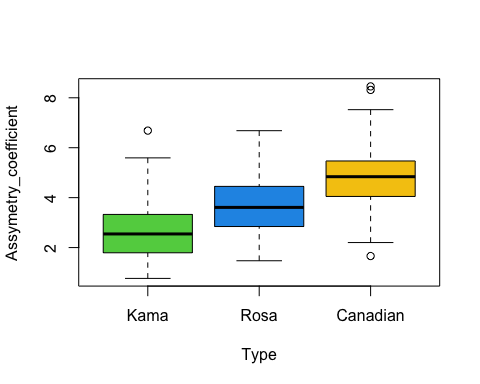
boxplot(Length\_of\_Kernel~Type,names = labels, col = c(3,4,7))



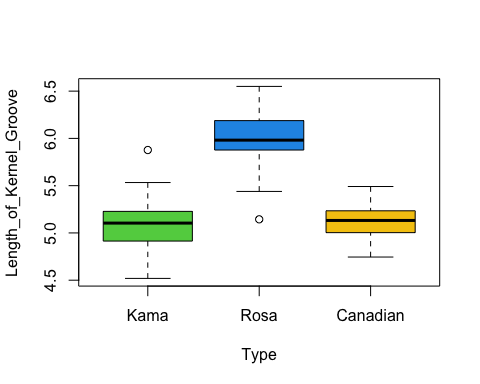
boxplot(Width\_of\_Kernel~Type,names = labels, col = c(3,4,7))



boxplot(Assymetry\_coefficient~Type,names = labels, col = c(3,4,7))



boxplot(Length\_of\_Kernel\_Groove~Type,names = labels, col = c(3,4,7))



From the boxplot it is clearly evident that Rosa wheat seeds are the largest among the mentioned categories while Canadian is the smallest in terms of geometrical properties. Kama wheat type looks neutral with respect to every variable.

Boxplot also showed that, besides being compact Canadian wheat seeds are highly symmetric in nature.

## Data Pre-prcoessing

### create a variable is\_kama and create a new data frame which have is\_kama as the qualitative variable except Type.

# Forming a new variable is\_Kama that will contain to values mainly 0 and 1.  
is\_kama <- wheat\_modified$Type  
is\_kama[is\_kama != 1] = 0  
  
## adding a new column is\_kama to the data frame.  
wheat\_modified <- cbind(wheat\_modified, is\_kama)  
  
# subsetting the data set to remove Type Column because we don't need it  
wheat\_modified <- subset(wheat\_modified, select = -Type)  
attach(wheat\_modified)

## The following object is masked \_by\_ .GlobalEnv:  
##   
## is\_kama

## The following objects are masked from wheatKernelDS:  
##   
## Area, Assymetry\_coefficient, Compactness, Length\_of\_Kernel,  
## Length\_of\_Kernel\_Groove, Perimeter, Width\_of\_Kernel

head(wheat\_modified)  
## converting the is\_kama variable as a factor variable  
wheat\_modified$is\_kama <- as.factor(wheat\_modified$is\_kama)  
str(wheat\_modified)

### divide the data set into training and testing data set.

set.seed(39)  
# Dividing into 70% training data and 30% testing data.  
training\_indexes <- sample(1:nrow(wheat\_modified),   
 nrow(wheat\_modified) \* 0.7)  
  
wheat\_training <- wheat\_modified[training\_indexes,]  
wheat\_testing <- wheat\_modified[-training\_indexes,]

## Method 1 - Logistic Regression

### Model building

### fit a logistic regression model with training data set for is\_kama variable

lr\_model1 <- glm(is\_kama~., data = wheat\_training, family = binomial)  
summary(lr\_model1)

From the summary we can see that, except width of Kernel, every geometrical property has some significance. Assymetry and length of kernel groove are highly significant.

The AIC value for this model is 33.412

### fit a second model with significant variables from first model.

lr\_model2 <- glm(is\_kama~Area+Perimeter+Compactness+Length\_of\_Kernel+  
 Assymetry\_coefficient+Length\_of\_Kernel\_Groove,   
 data = wheat\_training, family = binomial)  
summary(lr\_model2)

From the summary we can see that, every geometrical property in this model has some significance. Area, Compactness, Assymetry and length of kernel groove are highly significant.

The AIC value for this model is 31.479

### fit a third model with highly significant variables from second model.

lr\_model3 <- glm(is\_kama~Area+Compactness+  
 Assymetry\_coefficient+Length\_of\_Kernel\_Groove,   
 data = wheat\_training, family = binomial)  
summary(lr\_model3)

From the summary we can see that, every geometrical property in this model has some significance. Area, Assymetry and length of kernel groove are highly significant.

The AIC value for this model is 82.752

### fit a fourth model with highly significant variables from third model.

lr\_model4 <- glm(is\_kama~Area+  
 Assymetry\_coefficient+Length\_of\_Kernel\_Groove,   
 data = wheat\_training, family = binomial)  
summary(lr\_model4)

From the summary we can see that, every geometrical property in this model has some significance. Area, Assymetry and length of kernel groove are highly significant.

The AIC value for this model is 84.939

### Model validation

### Validate all those 4 models with cross validation by using K-Fold Cross validation technique.

##### Cross Validation  
dev.off()  
model <- c(1:4)  
cv\_errorKF= rep (0,4)  
cv\_errorKF[1] <- cv.glm(wheat\_training,lr\_model1, K=10)$delta[1]   
cv\_errorKF[2] <- cv.glm(wheat\_training,lr\_model2, K=10)$delta[1]   
cv\_errorKF[3] <- cv.glm(wheat\_training,lr\_model3, K=10)$delta[1]  
cv\_errorKF[4] <- cv.glm(wheat\_training,lr\_model4, K=10)$delta[1]  
  
cv\_errorKF  
plot(cv\_errorKF~model, type="b", col="green",  
 ylab = "Error rate", main = "10 Fold Cross Validation for models",  
 xaxt="n", xlim = c(1,4))   
axis(1, at = 1:4)

### Model evaluation

From the cross validation plot, we saw that model 2 has the least error rate and if we compare the AIC values, the lesser the AIC value more better the model. So, with this also model has least AIC value. Therefore, Model 2 could be the best model for this logistic regression.

### Model testing

Now, after finding the best model, we need to check the accuracy of the best model to ensure that the results are true for every data. To do this we do testing on the testing data and find its confusion matrix and misclassification rate in order to know how well the model is performing.

lr\_prediction1 <- predict(lr\_model2, newdata = wheat\_testing,   
 type = "response")  
## creating a prediction class for misclassification rate  
lr\_predicted\_class\_1 <- rep(0, nrow(wheat\_testing))  
   
lr\_predicted\_class\_1[lr\_prediction1 > 0.5] = 1  
   
## Calculating misclassification rate  
   
lr\_misclassification\_table1 <- table(lr\_predicted\_class\_1,  
 wheat\_testing$is\_kama)  
   
   
cat("Misclassification matrix: ")  
print(lr\_misclassification\_table1)  
   
lr\_misclassification\_rate1 <- ((lr\_misclassification\_table1[1,2] +   
 lr\_misclassification\_table1[2,1])   
 / sum(lr\_misclassification\_table1))  
lr\_misclassification\_rate1

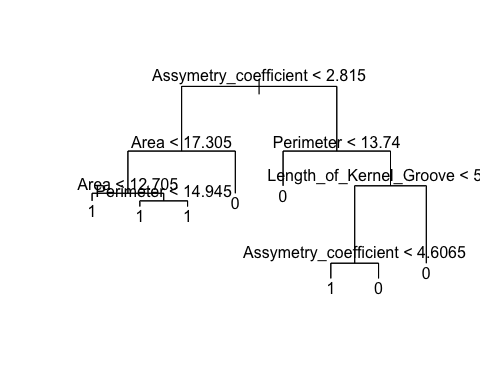
we can see that, the misclassification rate we got from model 2 is 0.04761905 which is approx. 4.76%. That means our model performed approx. 95.24% accurate result.

## Method 2 - Decision trees

### Model building

### create a full size decesion tree for the training data

class\_tree\_training <- tree(is\_kama~., data = wheat\_training)  
summary(class\_tree\_training)  
plot(class\_tree\_training,   
 main = "Classification tree on training data")  
text(class\_tree\_training, pretty = 0)



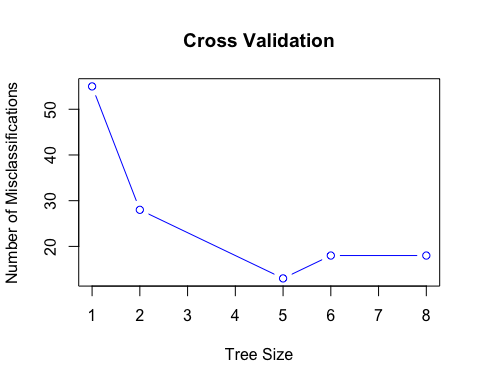
This decesion tree is formed on training data set for wheat side to identify the outcome and path to classify the “Kama” wheat seeds.

This decesion tree used only 4 variables which are “Assymetry\_coefficient”, “Area”, “Perimeter”, and “Length\_of\_Kernel\_Groove”. This tree has 8 terminal nodes to classify the “Kama” wheat type. Miclassification rate for this tree is 0.03401 which is 3.4%.

### Model Validation

### validate the tree with the help of cross validation

cv\_classtree <- cv.tree(class\_tree\_training, FUN = prune.misclass)  
cv\_classtree  
plot(cv\_classtree$size, cv\_classtree$dev, type = "b", xlab = "Tree Size",   
 ylab = "Number of Misclassifications",   
 main = "Cross Validation", col = "blue")



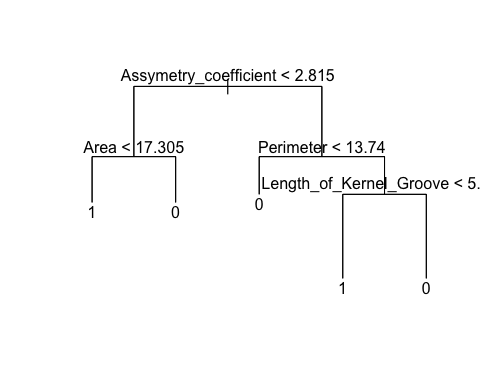
From the cross validaiton plot, it is evident that tree with size 5 has the less number of misclassifications. That means decision tree with size 5 should be the best model.

### Model Evaluation

### find the model of best size by pruning it

Now we got to know that tree size 5 is optimal. To get the specifications of the tree we need to prune the tree to get the resulatant tree with size 5.

pruned\_classtree <- prune.misclass(class\_tree\_training, best = 5)  
summary(pruned\_classtree)  
plot(pruned\_classtree, main = "Pruned tree with 5 terminal nodes")  
text(pruned\_classtree, pretty = 0)



Here we got the pruned tree, which has 5 terminal nodes and used 4 variables.

Our tree representation is like this: Terminal node 1: If “Asymmetry Coefficient” is less than 2.815 and “Area” is less than 17.305 then that seed can be classified as “Kama” seed. Terminal node 2: If “Asymmetry Coefficient” is less than 2.815 and “Area” is greater than 17.305 then that seed cannot be classified as “Kama” seed. Terminal node 3: If “Asymmetry Coefficient” is greater than 2.815 and “Perimeter” is less than 13.74 then that seed cannot be classified as “Kama” seed. Terminal node 4: If “Asymmetry Coefficient” is greater than 2.815 and “Perimeter” is greater than 13.74 and “Length of Kernel Groove” is less than 5.573 then that seed can be classified as “Kama” seed. Terminal node 5: If “Asymmetry Coefficient” is greater than 2.815 and “Perimeter” is greater than 13.74 and “Length of Kernel Groove” is greater than 5.573 then that seed cannot be classified as “Kama” seed.

### Model Testing

### predicting on testing data set to find the misclassification rate.

#### Testing the accuracy / Calculating misclassification rate  
  
pred\_tree2 <- predict(pruned\_classtree, newdata = wheat\_testing, type = "class")  
  
observed\_test <- wheat\_testing[, "is\_kama"]  
  
misclassification\_matrix\_pruned <- table(pred\_tree2, observed\_test)  
misclassification\_matrix\_pruned  
  
misclassrate\_pruned <- ( misclassification\_matrix\_pruned[1,2] + misclassification\_matrix\_pruned[2,1] )/ sum(misclassification\_matrix\_pruned)  
  
misclassrate\_pruned

from this, we can see that pruned tree obtained 14.28% misclassification rate. That means the accuracy from the final model of the decision tree is approx. 85.72%.

## Model Comparison

cat("Best Logic Regression Linear Model's Missclassification rate:"  
 , lr\_misclassification\_rate1\*100,"%\nBest Decision Tree's Missclassification rate:",   
 misclassrate\_pruned\*100,"%\n")

From this, we can see that logistic regression model is more better because it correctly classified more observation than decision trees in order to identify if a seed is “Kama” or not.

## Clustering

### Cluster Building

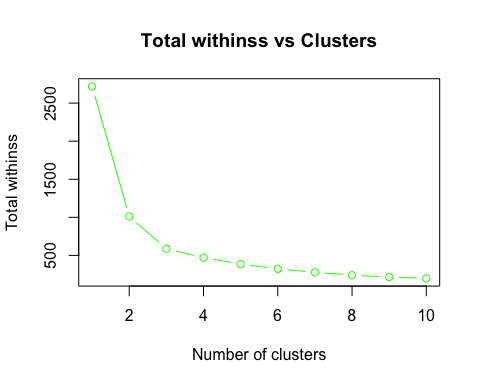
### cluster the data with 2,3 and 4 centers.

set.seed(39)  
wheat\_clustering <- wheat\_modified[,1:7]  
  
km\_cluster2 <- kmeans(wheat\_clustering, centers = 2, nstart = 20)  
km\_cluster2  
  
km\_cluster3 <- kmeans(wheat\_clustering, centers = 3, nstart = 20)  
km\_cluster3  
  
km\_cluster4 <- kmeans(wheat\_clustering, centers = 4, nstart = 20)  
km\_cluster4

### Cluster Evaluation

### find the best number of clusters by using their total.withinss value

clusters1 <- c(1:10)  
withiness1 <- c()  
  
for(i in clusters1){  
 tot\_with1 <- kmeans(wheat\_clustering, centers = i, nstart = 20)$tot.withinss  
 withiness1 <- c(withiness1, tot\_with1)  
}  
  
plot(withiness1~clusters1, type="b", col="green",  
 xlab = "Number of clusters", ylab = "Total withinss",  
 main = "Total withinss vs Clusters")



From this, we can see that clusters with 3 size is had a significant drop in total withinss and after that we can see the drop in the value but they are not that significant.

Therefore, selecting the best size of clusters as 3.

### visualise the clusters.

pca\_wheat <- prcomp(wheat\_modified[,1:7], scale. = TRUE)  
plot(pca\_wheat$x[,1:2], col = km\_cluster3$cluster + 1,  
 main = "Visualisation of cluster with 3 groups")  
legend("bottomleft",   
 legend=c("cluster 1","cluster 2", "cluster 3"), title="Clusters",  
 col = c(2, 3, 4), pch = 1)

