Iris-Classification Project

Henry Chan

January 2, 2020

Executive Summary

The Iris dataset was used in R.A. Fisher's classic 1936 paper, The Use of Multiple Measurements in Taxonomic Problems, and can also be found on the UCI Machine Learning Repository.

It includes three iris species with 50 samples each as well as some properties about each flower. One flower species is linearly separable from the other two, but the other two are not linearly separable from each other.

The columns in this dataset are:

1.Id - unique ID of the samples 2.SepalLengthCm - Length of the sepal (in cm) 3.SepalWidthCm - Width of the sepal (in cm) 4.PetalLengthCm - Length of the petal (in cm) 5.PetalWidthCm - Width of the petal (in cm) 6.Species - Species name

The aim of the project is to create a machine learning algorithm to predict the iris species correctly based on the given attributes.

Machine Learning Methods

Install Necessary Packages

```
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: caret
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
if(!require(tibble)) install.packages("tibble", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tibble
```

```
if(!require(dplyr)) install.packages("dplyr", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: dplyr
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Load dataset from csv file

```
data <- read.csv("iris.csv", header=TRUE)
```

Dataset summary

Dataset dimensions

```
dim(data)
```

```
## [1] 150   6
```

View headers and types of columns

```
sapply(data, class)
```

```
##            Id SepalLengthCm  SepalWidthCm PetalLengthCm  PetalWidthCm
##     "integer"     "numeric"     "numeric"     "numeric"     "numeric"
##       Species
##      "factor"
```

List of Species class levels

```
levels(data$Species)
```

```
## [1] "Iris-setosa"     "Iris-versicolor" "Iris-virginica"
```

Statistcal summary of dataset

```
summary(data)
```

```
##        Id         SepalLengthCm    SepalWidthCm   PetalLengthCm
##  Min.   :  1.00   Min.   :4.300   Min.   :2.000   Min.   :1.000
##  1st Qu.: 38.25   1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600
##  Median : 75.50   Median :5.800   Median :3.000   Median :4.350
##  Mean   : 75.50   Mean   :5.843   Mean   :3.054   Mean   :3.759
##  3rd Qu.:112.75   3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100
##  Max.   :150.00   Max.   :7.900   Max.   :4.400   Max.   :6.900
##   PetalWidthCm              Species
##  Min.   :0.100   Iris-setosa    :50
##  1st Qu.:0.300   Iris-versicolor:50
##  Median :1.300   Iris-virginica :50
##  Mean   :1.199
##  3rd Qu.:1.800
##  Max.   :2.500
```

Distribution of Species by frequency and percentage

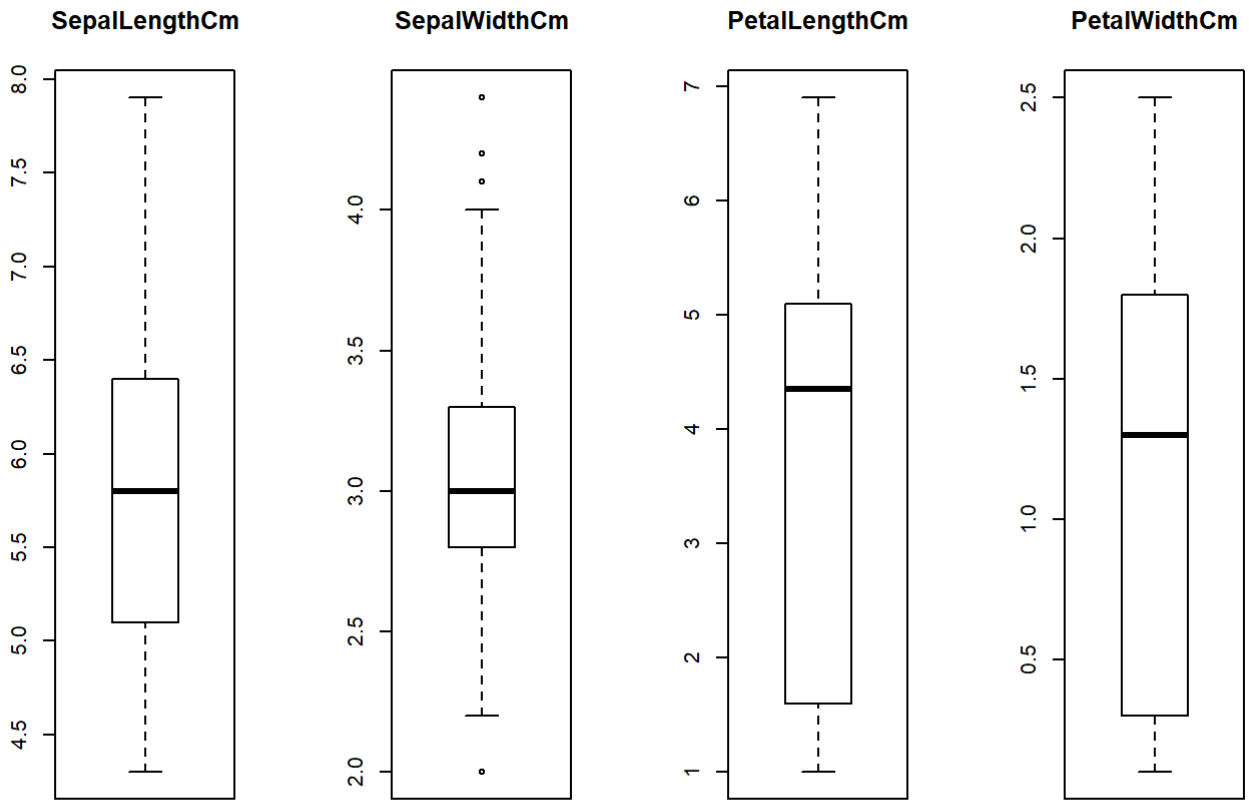
```
percentage <- prop.table(table(data$Species)) * 100
cbind(freq=table(data$Species), percentage=percentage)
```

```
##                 freq percentage
## Iris-setosa       50   33.33333
## Iris-versicolor   50   33.33333
## Iris-virginica    50   33.33333
```

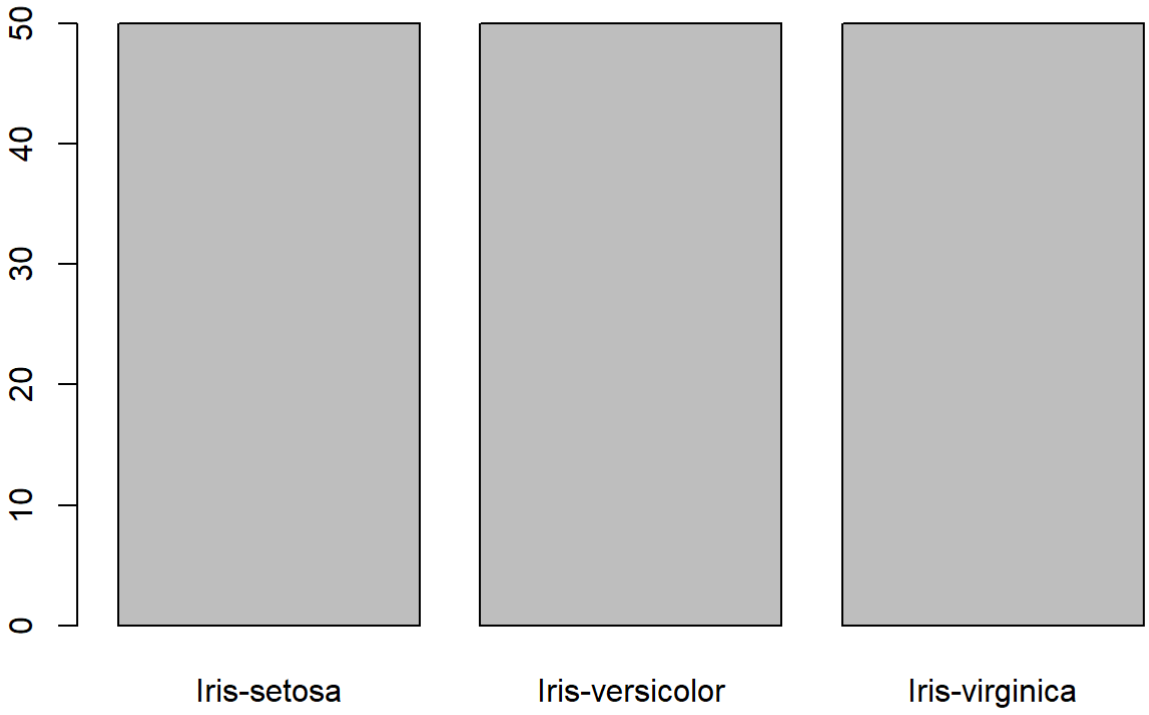
Split dataset into x and y, y being class labels

```
x <- data[,2:5]
y <- data[,6]
```

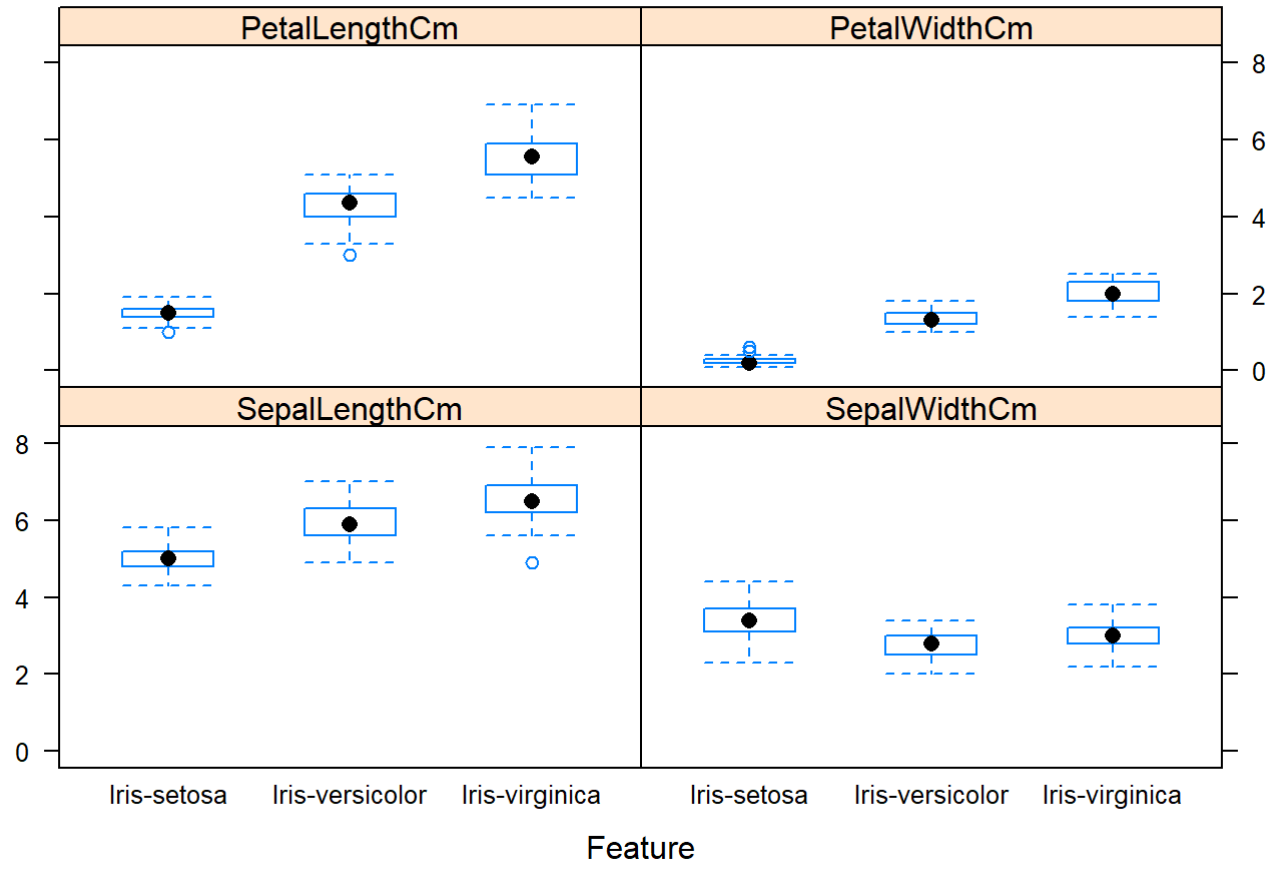
Boxplot for each attribute



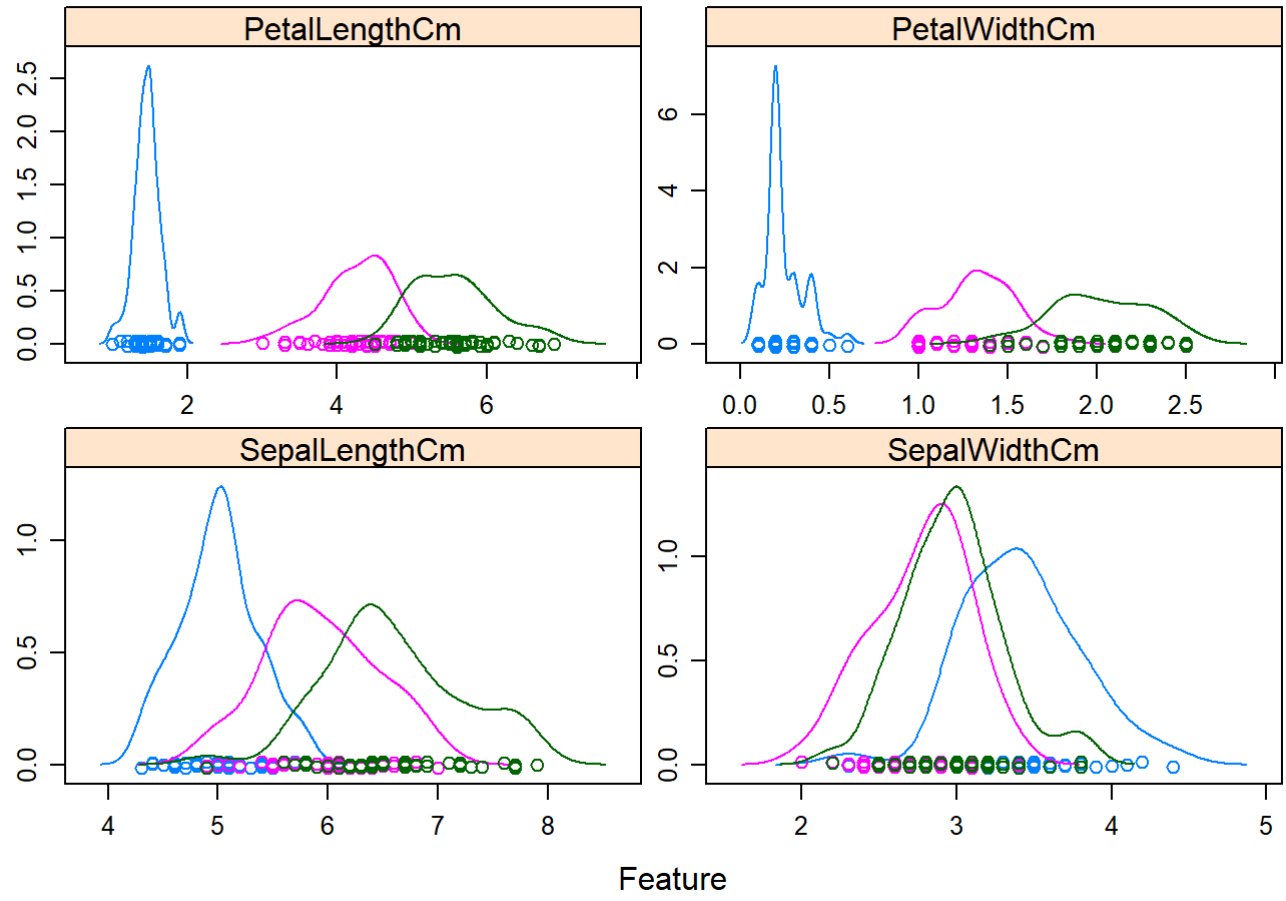
####Barplot showing frequency of each class



####Box and whisker plots by class for each attribute



####Density plots by class for each attribute



Machine Learning Model Building

Split the dataset into training and test set using createDataPartition(), 80% of data as training set and 20% of data as test set

```
test_index <- createDataPartition(data$Species, p = 0.8, list = FALSE)
train <- data[test_index,]
test <- data[-test_index,]
```

Algorithms will be assessed using 10-fold crossvalidation, setup here

```
control <- trainControl(method="cv", number=10)
metric <- "Accuracy"
```

5 machine learning models are introduced and respective accuracy of the prediction on test set are compared

Linear Discriminant Analysis

```
set.seed(1)
fit.lda <- train(Species~., data=data, method="lda", metric=metric, trControl=control)
predictions.lda <- predict(fit.lda,test)
```

Decision Tree

```
set.seed(1)
fit.rpart <- train(Species~., data=data, method="rpart", metric=metric, trControl=control)
predictions.rpart <- predict(fit.rpart,test)
```

k-Nearest Neighbors

```
set.seed(1)
fit.knn <- train(Species~., data=data, method="knn", metric=metric, trControl=control)
predictions.knn <- predict(fit.knn,test)
```

Support Vector Machines

```
set.seed(1)
fit.svm <- train(Species~., data=data, method="svmRadial", metric=metric, trControl=control)
predictions.svm <- predict(fit.svm,test)
```

Random Forest

```
set.seed(1)
fit.rf <- train(Species~., data=data, method="rf", metric=metric, trControl=control)
predictions.rf <- predict(fit.rf,test)
```

Results

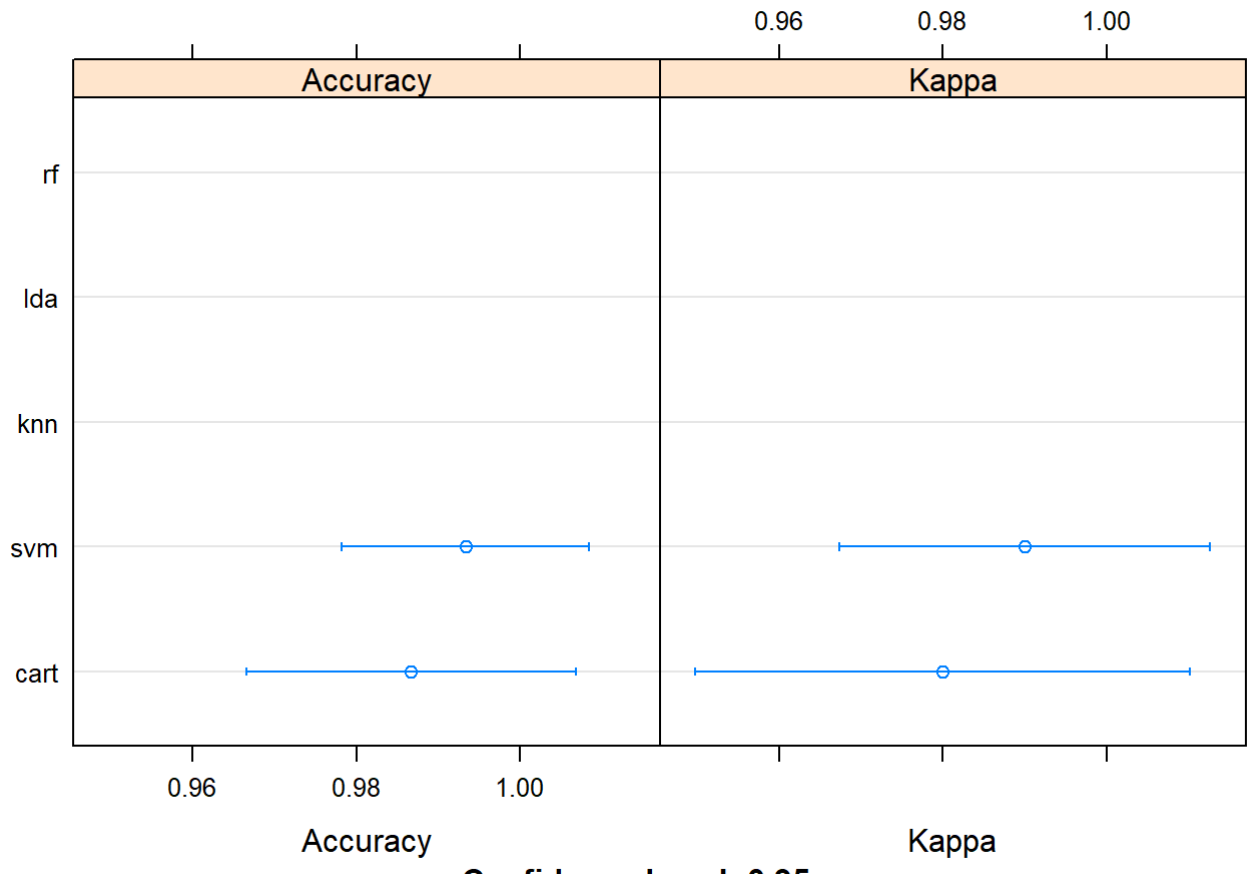
Summarize model accuracies

Summary of Accuracy and Kappa of different models

```
results <- resamples(list(lda=fit.lda, cart=fit.rpart, knn=fit.knn, svm=fit.svm, rf=fit.rf))
summary(results)
```

```
##
## Call:
## summary.resamples(object = results)
##
## Models: lda, cart, knn, svm, rf
## Number of resamples: 10
##
## Accuracy
##           Min. 1st Qu. Median      Mean 3rd Qu. Max. NA's
## lda  1.0000000       1      1 1.0000000       1    1    0
## cart 0.9333333       1      1 0.9866667       1    1    0
## knn  1.0000000       1      1 1.0000000       1    1    0
## svm  0.9333333       1      1 0.9933333       1    1    0
## rf   1.0000000       1      1 1.0000000       1    1    0
##
## Kappa
##      Min. 1st Qu. Median Mean 3rd Qu. Max. NA's
## lda   1.0       1      1 1.00       1    1    0
## cart  0.9       1      1 0.98       1    1    0
## knn   1.0       1      1 1.00       1    1    0
## svm   0.9       1      1 0.99       1    1    0
## rf    1.0       1      1 1.00       1    1    0
```

```
dotplot(results)
```



####Evaluate confustion matrix of the models' predictions on test data

```
confusionMatrix(predictions.lda, test$Species)
```

```
## Confusion Matrix and Statistics
##
##                 Reference
## Prediction        Iris-setosa Iris-versicolor Iris-virginica
##   Iris-setosa              10               0              0
##   Iris-versicolor           0              10              0
##   Iris-virginica            0               0             10
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.8843, 1)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 4.857e-15
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Iris-setosa Class: Iris-versicolor
## Sensitivity                      1.0000                 1.0000
## Specificity                      1.0000                 1.0000
## Pos Pred Value                   1.0000                 1.0000
## Neg Pred Value                   1.0000                 1.0000
## Prevalence                       0.3333                 0.3333
## Detection Rate                   0.3333                 0.3333
## Detection Prevalence             0.3333                 0.3333
## Balanced Accuracy                1.0000                 1.0000
##                      Class: Iris-virginica
## Sensitivity                         1.0000
## Specificity                         1.0000
## Pos Pred Value                      1.0000
## Neg Pred Value                      1.0000
## Prevalence                          0.3333
## Detection Rate                      0.3333
## Detection Prevalence                0.3333
## Balanced Accuracy                   1.0000
```

```
confusionMatrix(predictions.rpart, test$Species)
```

```
## Confusion Matrix and Statistics
##
##                 Reference
## Prediction        Iris-setosa Iris-versicolor Iris-virginica
##   Iris-setosa              10               0              0
##   Iris-versicolor           0              10              0
##   Iris-virginica            0               0             10
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.8843, 1)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 4.857e-15
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Iris-setosa Class: Iris-versicolor
## Sensitivity                      1.0000                 1.0000
## Specificity                      1.0000                 1.0000
## Pos Pred Value                   1.0000                 1.0000
## Neg Pred Value                   1.0000                 1.0000
## Prevalence                       0.3333                 0.3333
## Detection Rate                   0.3333                 0.3333
## Detection Prevalence             0.3333                 0.3333
## Balanced Accuracy                1.0000                 1.0000
##                      Class: Iris-virginica
## Sensitivity                         1.0000
## Specificity                         1.0000
## Pos Pred Value                      1.0000
## Neg Pred Value                      1.0000
## Prevalence                          0.3333
## Detection Rate                      0.3333
## Detection Prevalence                0.3333
## Balanced Accuracy                   1.0000
```

```
confusionMatrix(predictions.knn, test$Species)
```

```
## Confusion Matrix and Statistics
##
##                 Reference
## Prediction        Iris-setosa Iris-versicolor Iris-virginica
##   Iris-setosa              10               0              0
##   Iris-versicolor           0              10              0
##   Iris-virginica            0               0             10
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.8843, 1)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 4.857e-15
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Iris-setosa Class: Iris-versicolor
## Sensitivity                      1.0000                 1.0000
## Specificity                      1.0000                 1.0000
## Pos Pred Value                   1.0000                 1.0000
## Neg Pred Value                   1.0000                 1.0000
## Prevalence                       0.3333                 0.3333
## Detection Rate                   0.3333                 0.3333
## Detection Prevalence             0.3333                 0.3333
## Balanced Accuracy                1.0000                 1.0000
##                      Class: Iris-virginica
## Sensitivity                         1.0000
## Specificity                         1.0000
## Pos Pred Value                      1.0000
## Neg Pred Value                      1.0000
## Prevalence                          0.3333
## Detection Rate                      0.3333
## Detection Prevalence                0.3333
## Balanced Accuracy                   1.0000
```

```
confusionMatrix(predictions.svm, test$Species)
```

```
## Confusion Matrix and Statistics
##
##                 Reference
## Prediction        Iris-setosa Iris-versicolor Iris-virginica
##   Iris-setosa              10               0              0
##   Iris-versicolor           0              10              0
##   Iris-virginica            0               0             10
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.8843, 1)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 4.857e-15
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Iris-setosa Class: Iris-versicolor
## Sensitivity                      1.0000                 1.0000
## Specificity                      1.0000                 1.0000
## Pos Pred Value                   1.0000                 1.0000
## Neg Pred Value                   1.0000                 1.0000
## Prevalence                       0.3333                 0.3333
## Detection Rate                   0.3333                 0.3333
## Detection Prevalence             0.3333                 0.3333
## Balanced Accuracy                1.0000                 1.0000
##                      Class: Iris-virginica
## Sensitivity                         1.0000
## Specificity                         1.0000
## Pos Pred Value                      1.0000
## Neg Pred Value                      1.0000
## Prevalence                          0.3333
## Detection Rate                      0.3333
## Detection Prevalence                0.3333
## Balanced Accuracy                   1.0000
```

```
confusionMatrix(predictions.rf, test$Species)
```

```
## Confusion Matrix and Statistics
##
##                 Reference
## Prediction        Iris-setosa Iris-versicolor Iris-virginica
##   Iris-setosa              10               0              0
##   Iris-versicolor           0              10              0
##   Iris-virginica            0               0             10
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.8843, 1)
##     No Information Rate : 0.3333
##     P-Value [Acc > NIR] : 4.857e-15
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Iris-setosa Class: Iris-versicolor
## Sensitivity                      1.0000                 1.0000
## Specificity                      1.0000                 1.0000
## Pos Pred Value                   1.0000                 1.0000
## Neg Pred Value                   1.0000                 1.0000
## Prevalence                       0.3333                 0.3333
## Detection Rate                   0.3333                 0.3333
## Detection Prevalence             0.3333                 0.3333
## Balanced Accuracy                1.0000                 1.0000
##                      Class: Iris-virginica
## Sensitivity                         1.0000
## Specificity                         1.0000
## Pos Pred Value                      1.0000
## Neg Pred Value                      1.0000
## Prevalence                          0.3333
## Detection Rate                      0.3333
## Detection Prevalence                0.3333
## Balanced Accuracy                   1.0000
```

Create a table to summarise the accuracy of different models

Linear Discriminant Analysis

```
cm <- confusionMatrix(predictions.lda, test$Species)
overall <- cm$overall
overall.accuracy <- overall['Accuracy']

Summary <- tibble(Model = "lda", Accuracy = overall.accuracy)
```

Decision Tree

```
cm <- confusionMatrix(predictions.rpart, test$Species)
overall <- cm$overall
overall.accuracy <- overall['Accuracy']

Summary <- bind_rows(Summary,
                     tibble(Model="rpart",
                            Accuracy = overall.accuracy))
```

k-Nearest Neighbors

```
cm <- confusionMatrix(predictions.knn, test$Species)
overall <- cm$overall
overall.accuracy <- overall['Accuracy']

Summary <- bind_rows(Summary,
                     tibble(Model="knn",
                            Accuracy = overall.accuracy))
```

Support Vector Machines

```
cm <- confusionMatrix(predictions.svm, test$Species)
overall <- cm$overall
overall.accuracy <- overall['Accuracy']

Summary <- bind_rows(Summary,
                     tibble(Model="svm",
                            Accuracy = overall.accuracy))
```

Random Forest

```
cm <- confusionMatrix(predictions.rf, test$Species)
overall <- cm$overall
overall.accuracy <- overall['Accuracy']

Summary <- bind_rows(Summary,
                     tibble(Model="rf",
                            Accuracy = overall.accuracy))
```

Print summary table of models' accuracy

```
print(Summary)
```

```
## # A tibble: 5 x 2
##   Model Accuracy
##   <chr>    <dbl>
## 1 lda          1
## 2 rpart        1
## 3 knn          1
## 4 svm          1
## 5 rf           1
```

Conclusion

In the project, 5 models are introduced: Linear Discriminant Analysis (lda), Decision Tree (rpart), k-Nearest Neighbors (knn), Support Vector Machines (svm), Random Forest (rf). Algorithms are built based on train set data and are applied to test set for prediction. Accuracy of predictions of different models is summarised in the table. Based on the result, all models give 100% accuracy.