iris

Nguyen Thanh Tung

6/16/2019

Contents

1 Introduction

1.1 Describe the dataset

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper The use of multiple measurements in taxonomic problems as an example of linear discriminant analysis.[1] It is sometimes called Anderson's Iris data set because Edgar Anderson collected the data to quantify the morphologic variation of Iris flowers of three related species.[2] Two of the three species were collected in the Gaspé Peninsula "all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus".[3]

The data set consists of 50 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four features were measured from each sample: the length and the width of the sepals and petals, in centimeters. Based on the combination of these four features, Fisher developed a linear discriminant model to distinguish the species from each other.

For this challenge, I use "iris" data from R. "iris" is a data frame with 150 cases (rows) and 5 variables (columns) named Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, and Species.

```r
#Necessary Package for the challenge
library(caret)
```

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

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

```
## Registered S3 methods overwritten by 'ggplot2':
##   method         from
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
```

```r
library(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
```

```r
library(dslabs)
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```r
library(reshape)
```

```
##
## Attaching package: 'reshape'
```

```
## The following object is masked from 'package:dplyr':
##
##     rename
```

```r
library(pastecs)
```

```
##
## Attaching package: 'pastecs'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```r
#load data
data("iris")
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```r
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

1.2 Goal of the project

The goal of the project is buidling a machine learning model to predict the species based on given information: Sepal.Length, Sepal.Width, Petal.Length, Petal.Width.

The accuracy metric to measure the performance of the model is

$$Accuracy = \frac{Number\ of\ True\ prediction}{Number\ of\ Total\ prediction}$$

2 Analysis

2.1 Data exploration method & Insights collected

2.1.1 Summary statistic

#data exploration

- Summary statistic of variables:

```r
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100  
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300  
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300  
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199  
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  
##        Species
##  setosa    :50
##  versicolor:50
##  virginica :50
##
##
##
```

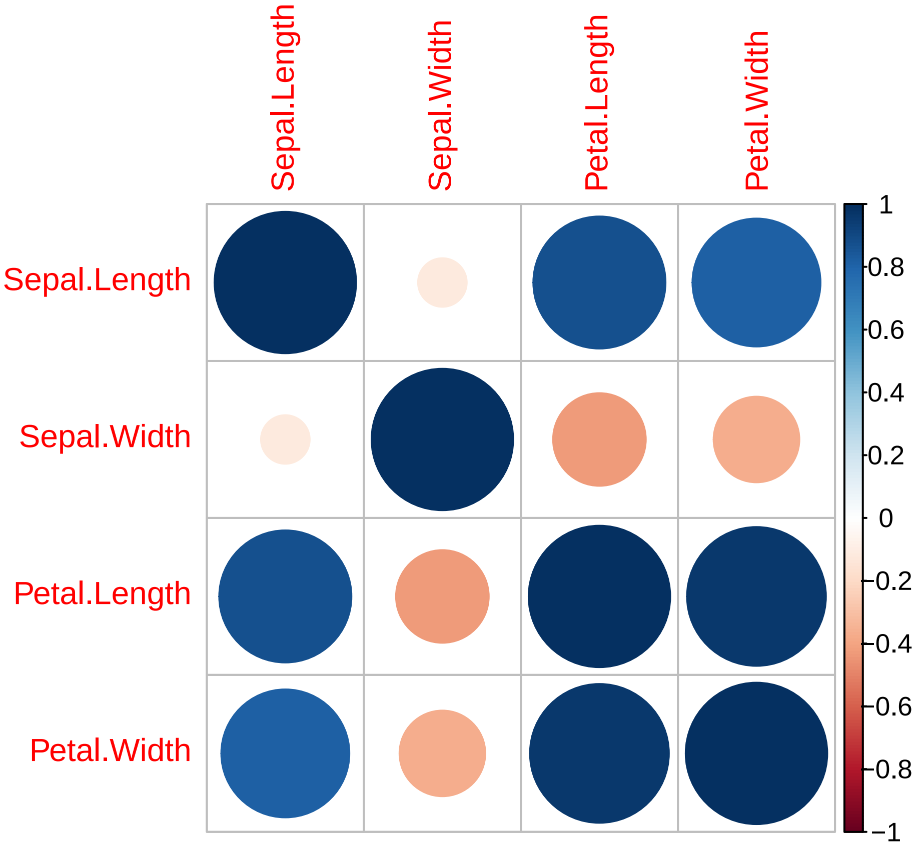
- From the summary statistic, we can see that the scale of variables is different. We should normalize data.

```
iris_norm <- data.frame("Species" = iris[,5] ,apply(iris[, 1:4], 2, function(x) (x - min(x))/(max(x)-mi
stat.desc(iris_norm)
```

```
##           Species Sepal.Length  Sepal.Width Petal.Length  Petal.Width
## nbr.val        NA 150.00000000 150.00000000 150.00000000 150.00000000
## nbr.null       NA   1.00000000   1.00000000   1.00000000   5.00000000
## nbr.na         NA   0.00000000   0.00000000   0.00000000   0.00000000
## min            NA   0.00000000   0.00000000   0.00000000   0.00000000
## max            NA   1.00000000   1.00000000   1.00000000   1.00000000
## range          NA   1.00000000   1.00000000   1.00000000   1.00000000
## sum            NA  64.30555556  66.08333333  70.11864407  68.70833333
## median         NA   0.41666667   0.41666667   0.56779661   0.50000000
## mean           NA   0.42870370   0.44055556   0.46745763   0.45805556
## SE.mean        NA   0.01878092   0.01482847   0.02442983   0.02593185
## CI.mean        NA   0.03711135   0.02930126   0.04827367   0.05124168
## var            NA   0.05290845   0.03298254   0.08952249   0.10086914
## std.dev        NA   0.23001837   0.18161095   0.29920309   0.31759903
## coef.var       NA   0.53654393   0.41223167   0.64006462   0.69336356
```

- Variable correlation, some variable are highly correlated, which may cause collinearity and lower the performance of the model

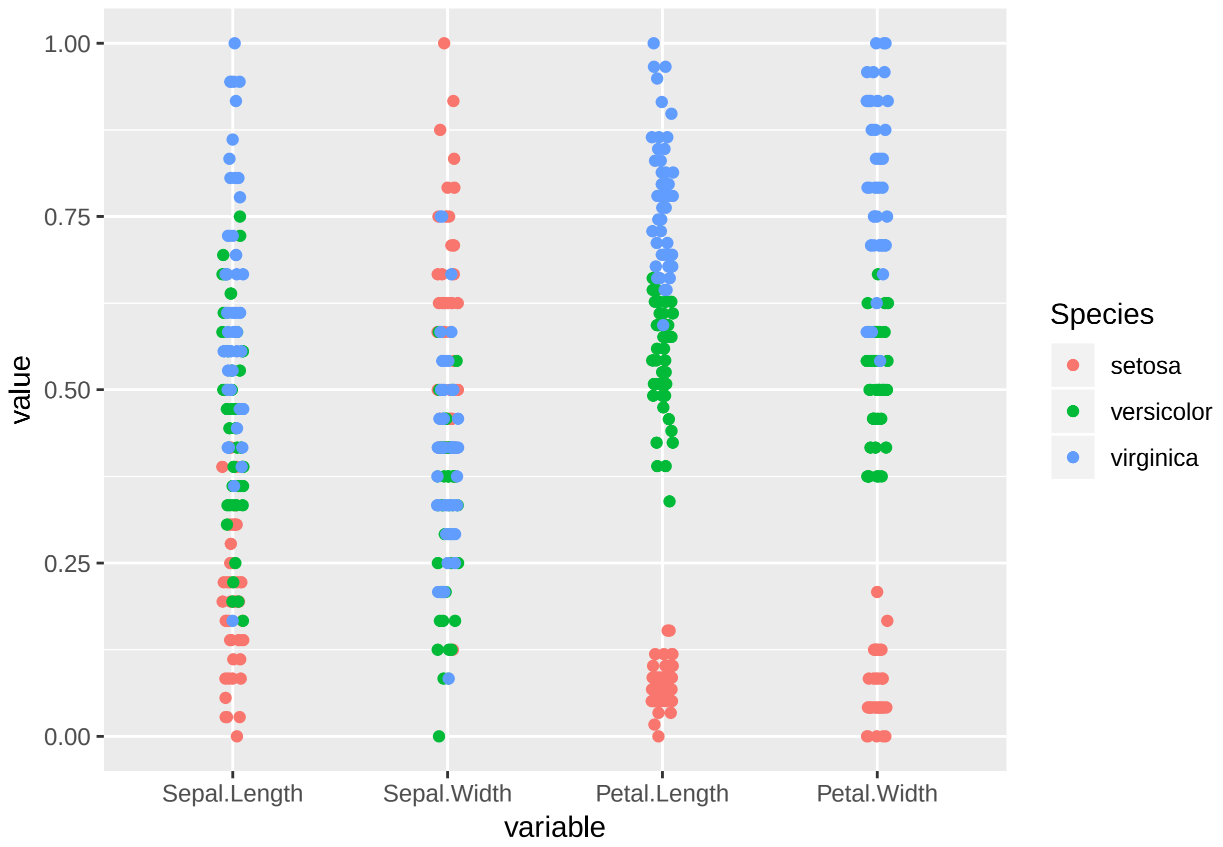
```
iris_exclude_species <- iris_norm %>%
  subset(select = -c(1))
corrplot(cor(iris_exclude_species))
```



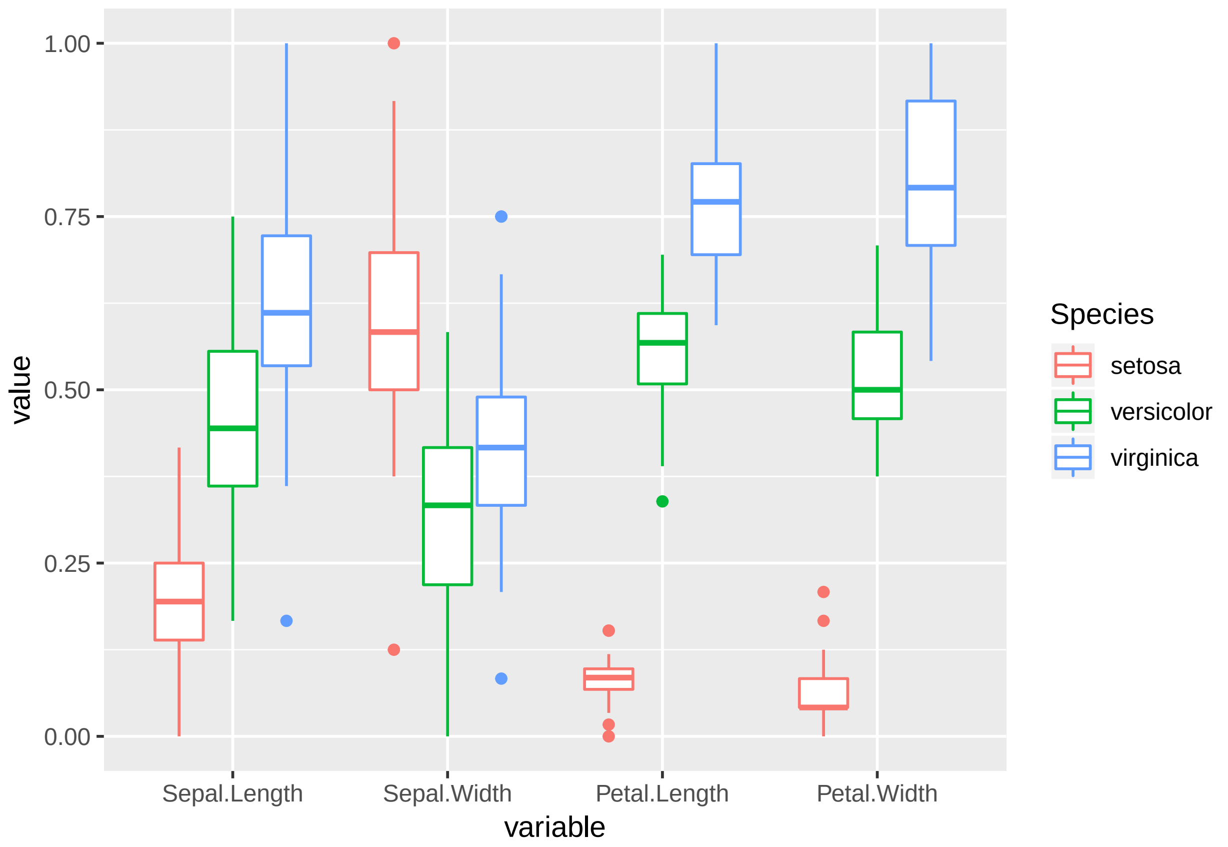
2.1.2 Visualization & insights collected

- Distribution of multiple varaibles with boxplot and scatter plot

```
iris_norm %>%
  melt(id = "Species") %>%
  ggplot(aes(x = variable, y = value, color = Species)) +
  geom_jitter(width = 0.05)
```



```
iris_norm %>%
  melt(id = "Species") %>%
  ggplot(aes(x = variable, y = value, color = Species)) +
  geom_boxplot()
```



+ Insight collected: Petal.Width and Sepal.Width are two variable which have high variace and low correlation

2.2 Modelling approach

2.2.1 Create train set & test set

```
#create train set & test set
index <- createDataPartition(iris$Species, times = 1, p = 0.8, list = FALSE)
train_set <- iris[index,]
test_set <- iris[-index,]
str(train_set)
```

```
## 'data.frame':    120 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.7 5 5.4 4.6 5 4.4 4.8 4.8 4.3 ...
##  $ Sepal.Width : num  3.5 3.2 3.6 3.9 3.4 3.4 2.9 3.4 3 3 ...
##  $ Petal.Length: num  1.4 1.3 1.4 1.7 1.4 1.5 1.4 1.6 1.4 1.1 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.2 0.1 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
str(test_set)
```

```
## 'data.frame':    30 obs. of  5 variables:
##  $ Sepal.Length: num  4.9 4.6 4.9 5.4 4.6 4.7 4.8 5.1 5 4.6 ...
##  $ Sepal.Width : num  3 3.1 3.1 3.7 3.6 3.2 3.1 3.4 3.5 3.2 ...
##  $ Petal.Length: num  1.4 1.5 1.5 1.5 1 1.6 1.6 1.5 1.6 1.4 ...
##  $ Petal.Width : num  0.2 0.2 0.1 0.2 0.2 0.2 0.2 0.2 0.6 0.2 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

2.2.2 Naive bayes method for top 2 predictor with highest variability and have low correlation

```
fit <- train(Species ~ Petal.Width + Sepal.Width, data = train_set, method = "nb")
fit
```

```
## Naive Bayes
##
## 120 samples
##   2 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 120, 120, 120, 120, 120, 120, ...
## Resampling results across tuning parameters:
##
##   usekernel  Accuracy   Kappa
##   FALSE      0.9683711  0.9512132
##    TRUE      0.9482428  0.9205201
##
## Tuning parameter 'fL' was held constant at a value of 0
## Tuning
##  parameter 'adjust' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were fL = 0, usekernel = FALSE
##  and adjust = 1.
```

```
varImp(fit)
```

```
## ROC curve variable importance
##
##   variables are sorted by maximum importance across the classes
##             setosa versicolor virginica
## Petal.Width 100.00     100.00       100
## Sepal.Width  59.92      59.92         0
```

```
y_hat <- predict(fit, newdata = test_set)

ac1 <- mean(y_hat == test_set$Species)
result <- data.frame("method" = "naive bayes top 2 predictor", "accuracy" = ac1)
result
```

```
##                        method  accuracy
## 1 naive bayes top 2 predictor 0.8666667
```

2.2.3 Naive bayes method for all predictor

```
fit <- train(Species ~ ., data = train_set, method = "nb")
fit
```

```
## Naive Bayes
##
## 120 samples
##   4 predictor
##   3 classes: 'setosa', 'versicolor', 'virginica'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 120, 120, 120, 120, 120, 120, ...
## Resampling results across tuning parameters:
##
##   usekernel  Accuracy   Kappa
##   FALSE      0.9370277  0.9043974
##    TRUE      0.9398790  0.9087088
##
## Tuning parameter 'fL' was held constant at a value of 0
## Tuning
##  parameter 'adjust' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were fL = 0, usekernel = TRUE
##  and adjust = 1.
```

```
varImp(fit)
```

```
## ROC curve variable importance
##
##   variables are sorted by maximum importance across the classes
##              setosa versicolor virginica
## Petal.Length 100.00     100.00    100.00
## Petal.Width  100.00     100.00    100.00
## Sepal.Length  87.23      61.10     87.23
## Sepal.Width   59.92      59.92      0.00
```

```
y_hat <- predict(fit, newdata = test_set)
ac2 <- mean(y_hat == test_set$Species)
result <- rbind(result,
                data.frame("method" = "naive bayes all predictor", "accuracy" = ac2))
result
```

```
##                         method  accuracy
## 1 naive bayes top 2 predictor 0.8666667
## 2   naive bayes all predictor 0.9333333
```

3 Result

Using simple method with just two predictor with high variace and low correlation provides very good result of 93% accuracy. Include all predictor only improve accuracy 3% to 96%.

4 Conclusion

The study has gone through 4 key steps: data processing, data exploration, modelling, result. The model Naive bayes method for all predictor provides good accuracy of 96%.