Homework 1

Ruth Colbert, Supriya Nunna, Alexander Lee, Harpreet Dhaliwal, Wenjuan Han, Arunabh Saikia

6/19/2021

For Knitting to pdf

```
r = getOption("repos")
r["CRAN"] = "http://cran.us.r-project.org"
options(repos = r)
```

Loading required packages

```
if(!require("pacman")) install.packages("pacman")
```

```
## Loading required package: pacman
```

```
pacman::p_load(esquisse, forecast, tidyverse,
               gplots, GGally, gganimate,
               mosaic, scales, mosaic, mapproj, mlbench, data.table)

library("corrplot")
```

```
## corrplot 0.89 loaded
```

```
search()
```

```
##  [1] ".GlobalEnv"         "package:corrplot"   "package:data.table"
##  [4] "package:mlbench"    "package:mapproj"    "package:maps"
##  [7] "package:scales"     "package:mosaic"     "package:ggridges"
## [10] "package:mosaicData" "package:ggformula"  "package:ggstance"
## [13] "package:Matrix"     "package:lattice"    "package:gganimate"
## [16] "package:GGally"     "package:gplots"     "package:forcats"
## [19] "package:stringr"    "package:dplyr"      "package:purrr"
## [22] "package:readr"      "package:tidyr"      "package:tibble"
## [25] "package:ggplot2"    "package:tidyverse"  "package:forecast"
## [28] "package:esquisse"   "package:pacman"     "package:stats"
## [31] "package:graphics"   "package:grDevices"  "package:utils"
## [34] "package:datasets"   "package:methods"    "Autoloads"
## [37] "package:base"
```

```
theme_set(theme_classic())
```

Loading Iris dataset

```
data("iris")
#checking class of iris dataset
class("iris")
```

```
## [1] "character"
```

```
#Converting to data.table
Iris.dt<-setDT(iris)
class(Iris.dt)
```

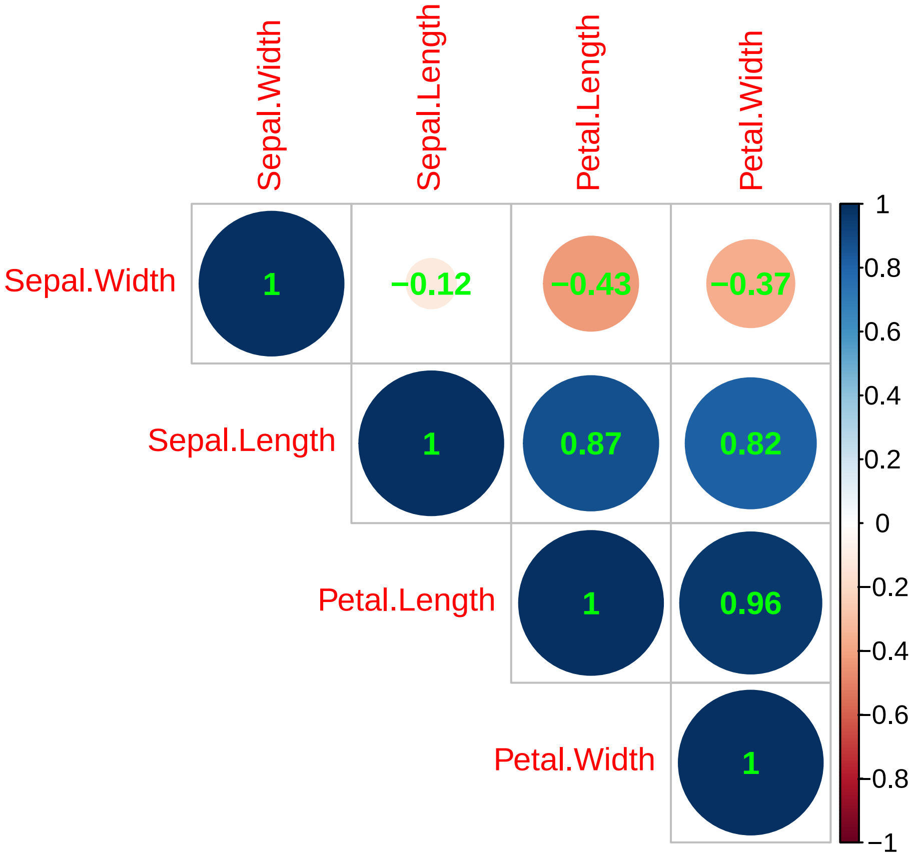
```
## [1] "data.table" "data.frame"
```

Question 1

```
#Keeping only numeric variables
Irisnum.dt <- Filter(is.numeric, Iris.dt)

#computing correlation of numeric variables
Iriscorr<-cor(Irisnum.dt)

#Visualizing the correlation matrix
Iriscorrplot<-corrplot(Iriscorr, type="upper", order="hclust",method="circle",addCoef.col = "Green")
```



As per the above plot, petal length and petal width has the highest correlation coefficient of 0.96 indicating strong positive linear relationship.

Question 2

1) Correlation coefficients do not provide causation for the relationship. Let's say, in Seattle, we found that winter coat sales are positively correlated to depression rate. However, winter coat sales are no way related to depression rate. This correlation could be caused by an unknown variable called "temperature" which affects both. When temperature decreases, the sales of winter coat increases. Also when the temperature decreases, people tend to be more depressed thus having high depression rate. This proves that Correlation does not provide causation.

2) In addition, the correlation coefficient only looks at linear relationships. For example, a vehicle requires gasoline to drive. The correlation is positive for high volume of gasoline to duration and/or distance the vehicle drives. However, this does not take into account the type of gas or type of car being driven. The correlation for one brand of car with 15 gallons of gas may be stronger due it driving further, but another brand of car might have have 20 gallons of gas and drive a less distance due to the type of gas and MPG. The addition of variables have a significant impact on the relationship but would not be identified in a correlation coefficient.

Question 3

```
#Keeping only numeric variables
Irisnum.dt <- Filter(is.numeric, Iris.dt)

#Calculating mean
Irismean<-sapply(Irisnum.dt, mean)
Irismean
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width 
##     5.843333     3.057333     3.758000     1.199333
```

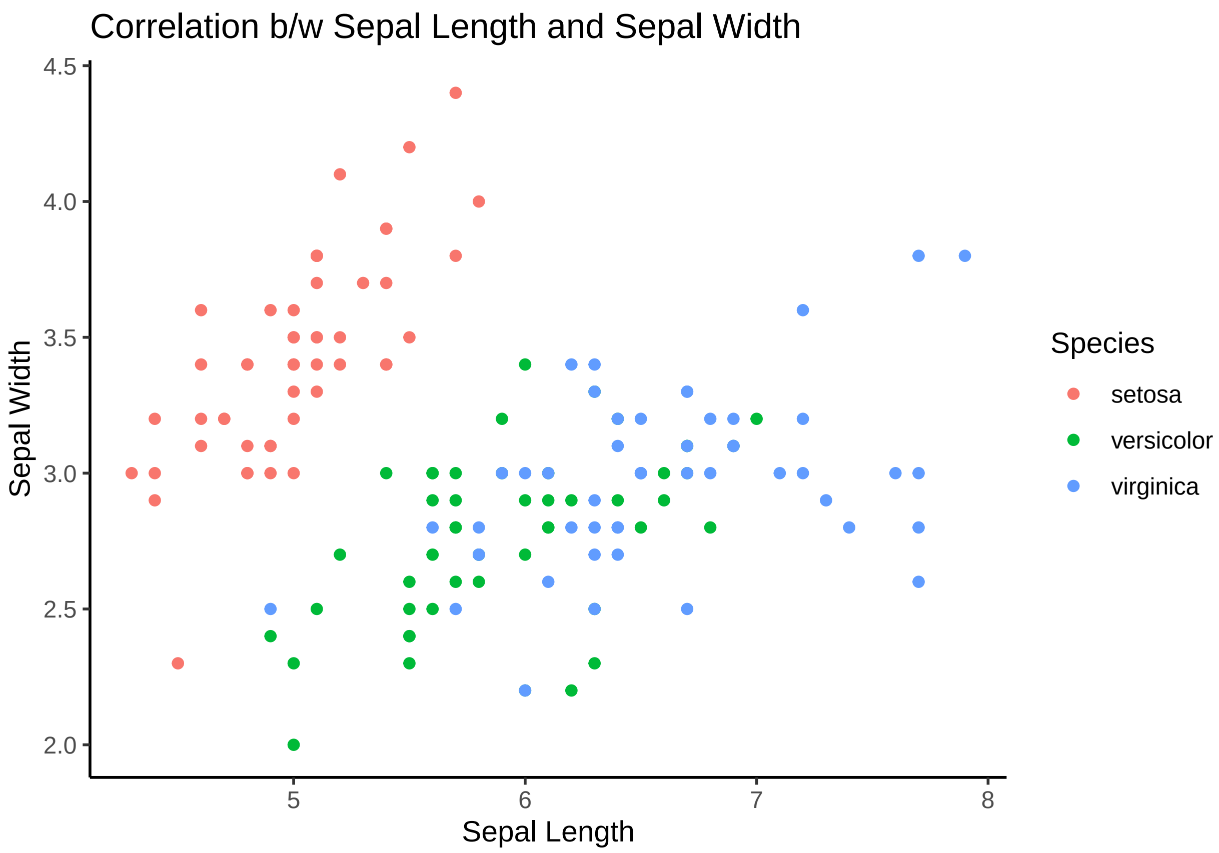
```
#Checking maximum of calculated mean
max(Irismean)
```

```
## [1] 5.843333
```

Sepal.length variable has the highest mean of 5.843333

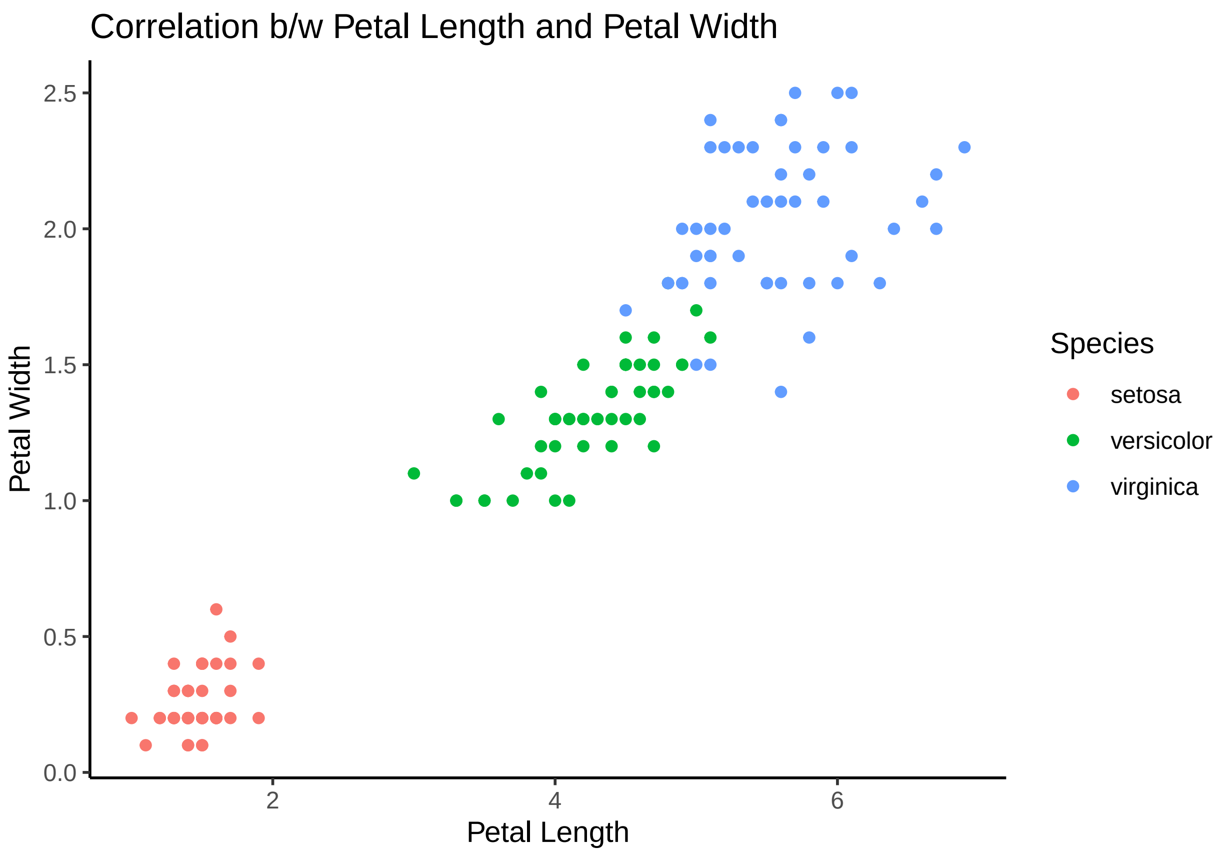
Question 4

```
ggplot(Iris.dt, aes(x=Sepal.Length, y=Sepal.Width, color=Species)) +
  geom_point()+
ggtitle("Correlation b/w Sepal Length and Sepal Width")+
   ylab("Sepal Width")+
  xlab("Sepal Length")
```



Question 5

```
ggplot(Iris.dt, aes(x=Petal.Length, y=Petal.Width, color=Species)) +
  geom_point()+
  ggtitle("Correlation b/w Petal Length and Petal Width")+
  ylab("Petal Width")+
  xlab("Petal Length")
```



Question 6

The Petal Length and Petal Width combination provides a better separation of records between the two graphs because the variables of Petal Length and Petal Width are more highly correlated (0.96) than Sepal Length and Width (-0.12). The Petal graph in question 5 represents lower error rate when calculating the relationship using predictive variable between petal length and petal width, resulting in more defined clusters per species. There is significant overlap between Versicolor and Virginica species in the Sepal graph in question 4. The high error rate in this graph makes it difficult to classify the species (low separation) using these specific variables when compared to Petal graph.

Question 7

```
(2+5)/100 #overall error rate
```

```
## [1] 0.07
```

```
1-.07 #accuracy percentage
```

```
## [1] 0.93
```

As per above calculations, the accuracy of the model is 93%.

NIR:

```
Setosa <- (45+5); Setosa #Sum of Actual Classifications of Setosa
```

```
## [1] 50
```

```
Versicolor <-(2+48); Versicolor #Sum of Actual Classifications of Versicolor
```

```
## [1] 50
```

The No Information Rate talks about the proportion of the dominant class. Calculating the actual classifications of each species, we gather the sum of each species and found they are equal [Versicolor = 50, Setosa = 50]. This means our NIR is .50.

Comparing NIR and accuracy:

Comparing the NIR (0.50) and the accuracy percentage of (0.93), the accuracy of the model is greater than the NIR. Since in NIR (0.5), only the dominant class is taken into account, our prediction will be correct only 50% of the time. Therefore we can conclude that the current model which has accuracy rate of 93% can predict the classifications 43% better than the NIR.

Question 8

```
Sensitivity <- 45 / (45 + 5);Sensitivity
```

```
## [1] 0.9
```

```
#Sensitivity = (TP / (TP + FN)) = 45 / (45 + 5) = 45 / 50 -> 0.9 (90%)
```

Question 9

```
Specificity <- 48 / (48 + 2);Specificity
```

```
## [1] 0.96
```

```
#Specificity = (TN / (TN + FP)) = 48 / (48 + 2) = 48 / 50 -> 0.96 (96%)
```

Question 10

```
Precision <- 45 / (45 + 2);Precision
```

```
## [1] 0.9574468
```

```
#Precision = (TP / (TP + FP)) = 45 / (45 + 2) = 45 / 47 -> 0.9574 (95.74%)
```