Iris Data Analysis

Charles Gonzalez-Dyer

## Setting Up My Environment

First I will load the needed packages and ensure that the data set was imported properly with the View() function.

install.packages("skimr")

##   
## The downloaded binary packages are in  
## /var/folders/xh/kbyxrc692q1gg4j15r1q22nw0000gn/T//Rtmp06Ihe7/downloaded\_packages

library(skimr)  
install.packages("caret")

##   
## The downloaded binary packages are in  
## /var/folders/xh/kbyxrc692q1gg4j15r1q22nw0000gn/T//Rtmp06Ihe7/downloaded\_packages

library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(datasets)  
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

data("iris")  
View(iris)

This data set contains measurements of iris flowers. To help me get a better idea of each variable in the data frame, I can use summary(*) and skim(*) to view general descriptive statistics.

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   
##   
##   
##

skim(iris)

Data summary

|  |  |
| --- | --- |
| Name | iris |
| Number of rows | 150 |
| Number of columns | 5 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 1 |
| numeric | 4 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| Species | 0 | 1 | FALSE | 3 | set: 50, ver: 50, vir: 50 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sepal.Length | 0 | 1 | 5.84 | 0.83 | 4.3 | 5.1 | 5.80 | 6.4 | 7.9 | ▆▇▇▅▂ |
| Sepal.Width | 0 | 1 | 3.06 | 0.44 | 2.0 | 2.8 | 3.00 | 3.3 | 4.4 | ▁▆▇▂▁ |
| Petal.Length | 0 | 1 | 3.76 | 1.77 | 1.0 | 1.6 | 4.35 | 5.1 | 6.9 | ▇▁▆▇▂ |
| Petal.Width | 0 | 1 | 1.20 | 0.76 | 0.1 | 0.3 | 1.30 | 1.8 | 2.5 | ▇▁▇▅▃ |

I will also check for null values.

sum(is.na(iris))

## [1] 0

Since there are no null values, I can proceed without needing to clean the data.

# Data Exploration

The first thing that I will want to investigate are the differences between the three different species of iris flower. To begin this process, I will create a pipeline to group the different species of flower before I skim().

iris %>%  
 dplyr::group\_by(Species) %>%  
 skim()

Data summary

|  |  |
| --- | --- |
| Name | Piped data |
| Number of rows | 150 |
| Number of columns | 5 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| numeric | 4 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | Species |

**Variable type: numeric**

| skim\_variable | Species | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sepal.Length | setosa | 0 | 1 | 5.01 | 0.35 | 4.3 | 4.80 | 5.00 | 5.20 | 5.8 | ▃▃▇▅▁ |
| Sepal.Length | versicolor | 0 | 1 | 5.94 | 0.52 | 4.9 | 5.60 | 5.90 | 6.30 | 7.0 | ▂▇▆▃▃ |
| Sepal.Length | virginica | 0 | 1 | 6.59 | 0.64 | 4.9 | 6.23 | 6.50 | 6.90 | 7.9 | ▁▃▇▃▂ |
| Sepal.Width | setosa | 0 | 1 | 3.43 | 0.38 | 2.3 | 3.20 | 3.40 | 3.68 | 4.4 | ▁▃▇▅▂ |
| Sepal.Width | versicolor | 0 | 1 | 2.77 | 0.31 | 2.0 | 2.52 | 2.80 | 3.00 | 3.4 | ▁▅▆▇▂ |
| Sepal.Width | virginica | 0 | 1 | 2.97 | 0.32 | 2.2 | 2.80 | 3.00 | 3.18 | 3.8 | ▂▆▇▅▁ |
| Petal.Length | setosa | 0 | 1 | 1.46 | 0.17 | 1.0 | 1.40 | 1.50 | 1.58 | 1.9 | ▁▃▇▃▁ |
| Petal.Length | versicolor | 0 | 1 | 4.26 | 0.47 | 3.0 | 4.00 | 4.35 | 4.60 | 5.1 | ▂▂▇▇▆ |
| Petal.Length | virginica | 0 | 1 | 5.55 | 0.55 | 4.5 | 5.10 | 5.55 | 5.88 | 6.9 | ▃▇▇▃▂ |
| Petal.Width | setosa | 0 | 1 | 0.25 | 0.11 | 0.1 | 0.20 | 0.20 | 0.30 | 0.6 | ▇▂▂▁▁ |
| Petal.Width | versicolor | 0 | 1 | 1.33 | 0.20 | 1.0 | 1.20 | 1.30 | 1.50 | 1.8 | ▅▇▃▆▁ |
| Petal.Width | virginica | 0 | 1 | 2.03 | 0.27 | 1.4 | 1.80 | 2.00 | 2.30 | 2.5 | ▂▇▆▅▇ |

Since the output is a bit cluttered, I will filter down the results a bit.

iris %>%  
 select(Sepal.Length, Sepal.Width, Species) %>%   
 dplyr::group\_by(Species) %>%  
 skim()

Data summary

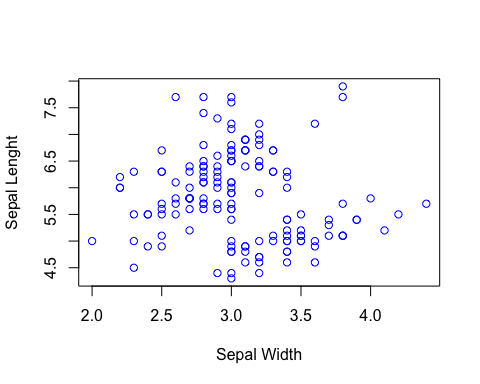
|  |  |
| --- | --- |
| Name | Piped data |
| Number of rows | 150 |
| Number of columns | 3 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| numeric | 2 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | Species |

**Variable type: numeric**

| skim\_variable | Species | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sepal.Length | setosa | 0 | 1 | 5.01 | 0.35 | 4.3 | 4.80 | 5.0 | 5.20 | 5.8 | ▃▃▇▅▁ |
| Sepal.Length | versicolor | 0 | 1 | 5.94 | 0.52 | 4.9 | 5.60 | 5.9 | 6.30 | 7.0 | ▂▇▆▃▃ |
| Sepal.Length | virginica | 0 | 1 | 6.59 | 0.64 | 4.9 | 6.23 | 6.5 | 6.90 | 7.9 | ▁▃▇▃▂ |
| Sepal.Width | setosa | 0 | 1 | 3.43 | 0.38 | 2.3 | 3.20 | 3.4 | 3.68 | 4.4 | ▁▃▇▅▂ |
| Sepal.Width | versicolor | 0 | 1 | 2.77 | 0.31 | 2.0 | 2.52 | 2.8 | 3.00 | 3.4 | ▁▅▆▇▂ |
| Sepal.Width | virginica | 0 | 1 | 2.97 | 0.32 | 2.2 | 2.80 | 3.0 | 3.18 | 3.8 | ▂▆▇▅▁ |

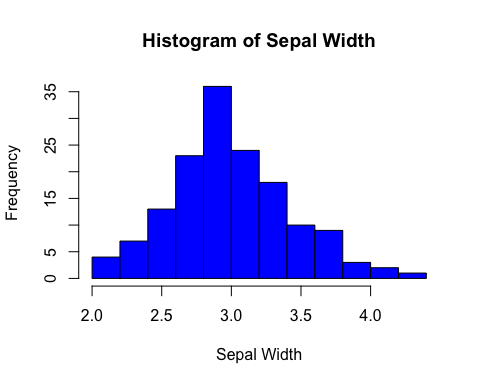
To get a deeper understanding of the data I can create a basic scatter plot.

plot(iris$Sepal.Width, iris$Sepal.Length, col="blue",  
 xlab = "Sepal Width", ylab = "Sepal Lenght")

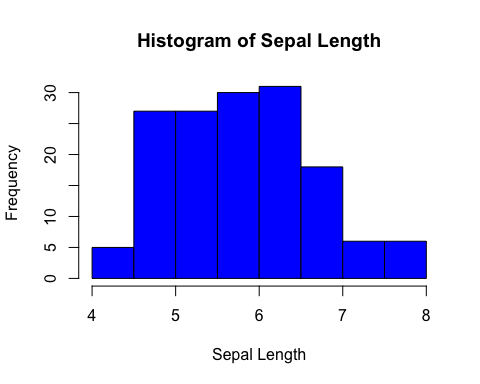


I can also create a histogram

hist(iris$Sepal.Width, col = "blue",  
 main= "Histogram of Sepal Width",  
 xlab= "Sepal Width")

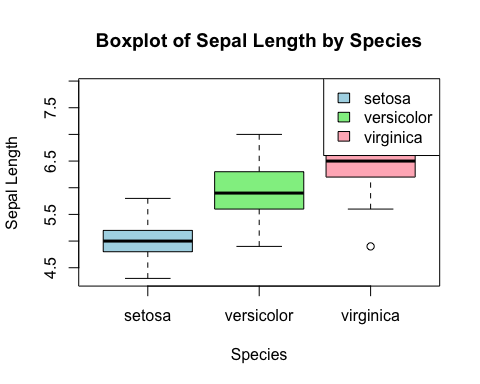


hist(iris$Sepal.Length, col = "blue",  
 main= "Histogram of Sepal Length",  
 xlab= "Sepal Length")



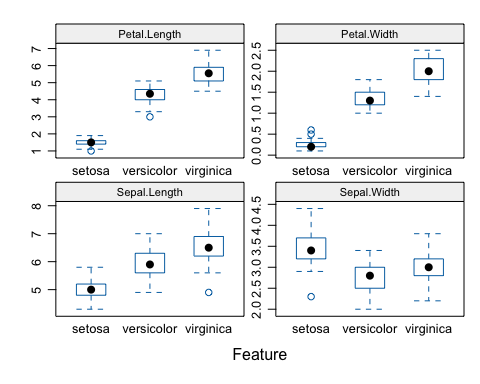
or even a box plot of sepal lengths for each species

boxplot(Sepal.Length ~ Species, data = iris,  
 main = "Boxplot of Sepal Length by Species",  
 xlab = "Species",  
 ylab = "Sepal Length",  
 col = c("lightblue", "lightgreen", "lightpink"))  
legend("topright", legend = levels(iris$Species),   
 fill = c("lightblue", "lightgreen", "lightpink"))



because of the numerous packages available for R there are even more ways to create similar plots!

featurePlot(x = iris[,1:4],  
 y = iris$Species,  
 plot = "box",  
 strip=strip.custom(par.strip.text=list(cex=0.7)),  
 scales = list(x = list(relation="free"),  
 y = list(relation= "free")))



## Statistical Analysis

After all of these descriptive statistics and visualizations it is time to run some statistical tests! I’ll start by determining if there is a statistically significant difference in the Sepal Length between the setosa and versicolor species. To start, I’ll create a new data frame that only contains the sepal lengths of setosa and versicolor.

df1 <- iris %>%   
 select(Sepal.Length, Species) %>%   
 filter(Species == "setosa" |  
 Species == "versicolor")

Having this new data frame will make running the t-test easier.

t.test(data = df1, Sepal.Length ~ Species)

##   
## Welch Two Sample t-test  
##   
## data: Sepal.Length by Species  
## t = -10.521, df = 86.538, p-value < 2.2e-16  
## alternative hypothesis: true difference in means between group setosa and group versicolor is not equal to 0  
## 95 percent confidence interval:  
## -1.1057074 -0.7542926  
## sample estimates:  
## mean in group setosa mean in group versicolor   
## 5.006 5.936

The output shows that the p-value is < 2.2e-16, far less than 0.05. This indicates that the difference in sepal length *is* significantly different between the two groups. Lastly, I will create a linear regression model examining the relationship between sepal length and sepal width.

summary(lm(Sepal.Width ~ Sepal.Length, data=iris))

##   
## Call:  
## lm(formula = Sepal.Width ~ Sepal.Length, data = iris)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.1095 -0.2454 -0.0167 0.2763 1.3338   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.41895 0.25356 13.48 <2e-16 \*\*\*  
## Sepal.Length -0.06188 0.04297 -1.44 0.152   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4343 on 148 degrees of freedom  
## Multiple R-squared: 0.01382, Adjusted R-squared: 0.007159   
## F-statistic: 2.074 on 1 and 148 DF, p-value: 0.1519

In the above model the **response variable** is sepal width and the **predictor variable** is sepal length. The p-value of 0.152 indicates that there is not enough evidence to conclude a significant linear relationship between Sepal.Width and Sepal.Length.

ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width)) +  
 geom\_point(color = "blue") +  
 geom\_smooth(method = "lm", se = FALSE, col = "red") +  
 labs(x = "Sepal Length",   
 y = "Sepal Width",   
 title = "Regression Model")+  
 geom\_segment(aes(x = 6.5, y = 4.0, xend = 5.5, yend = 3.1),   
 arrow = arrow(length = unit(0.2, "inches")),  
 color = "black") +  
 annotate("text", x = 6.5, y = 4.1, label = "Regression Line",   
 color = "black", hjust = 0)

## `geom\_smooth()` using formula = 'y ~ x'

