# Week 4 Assignment

By: Zach Adair

Regis University

MSDS 662: Exploratory Data Analysis

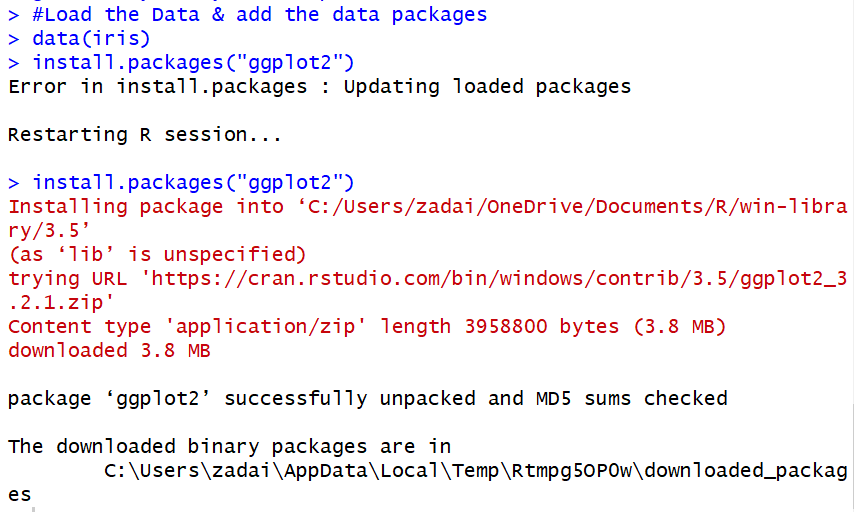
September 18th, 2019

## Introduction

An important step in the Exploratory Data Analysis process is the use of data visualizations to help analysts have an easier time with pulling information from big data by making it easy to understand and showing trends and volume disparities in pictorial form. In RStudio a great way to do this is with the command library of ggplot2, which is used to help take the generic visualizations a standard R analysis would do and enhances them, by adding more color and description to them which can help point out nuances in the data that could be used as important information. In the assignment for this week, I will use the ggplot2 package and analyze several RStudio data sources, with the intention of understanding the capabilities of ggplot2 while also being able to bring to light some information about these datasets you can’t find in a typical tabular form.

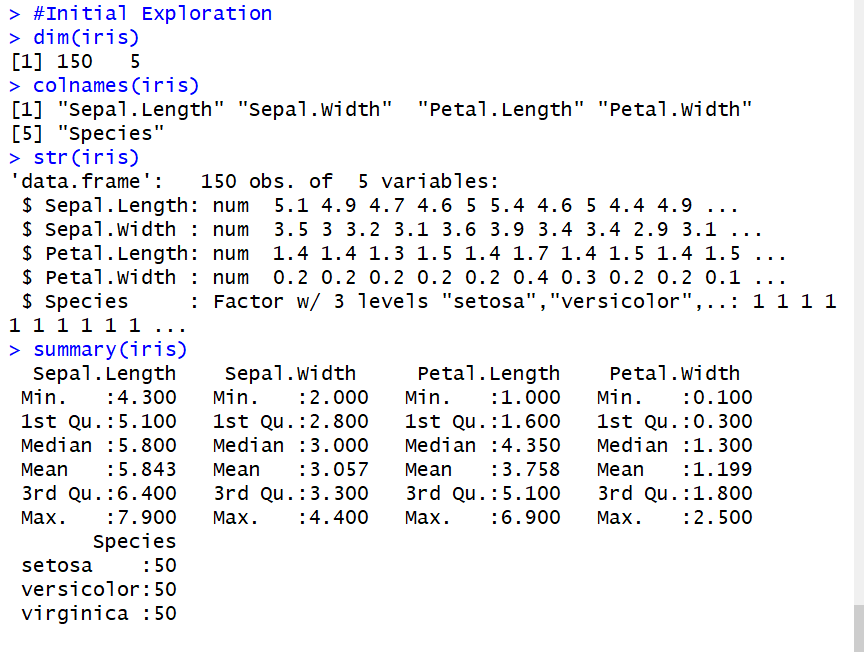
## Load data

Like what I would do for any analysis, I will need to load my data and my command packages which will be used. Which for this assignment, will be the iris dataset and ggplot2.



## Initial Analysis

Like with any analysis, it is smart to do some preliminary data exploration before plotting the data. First, I need to see what the rows and columns look like, what’s the structure like, and what a quick summary of the data might uncover.



As we can see above, the dimensions of the dataset are 150 rows and 5 variables. The Species column is the only non-numeric variable and also has 3 levels, all with the same number of entries.

With a little information about the dataset uncovered, we can now look at what kind of visualizations will come from this dataset.

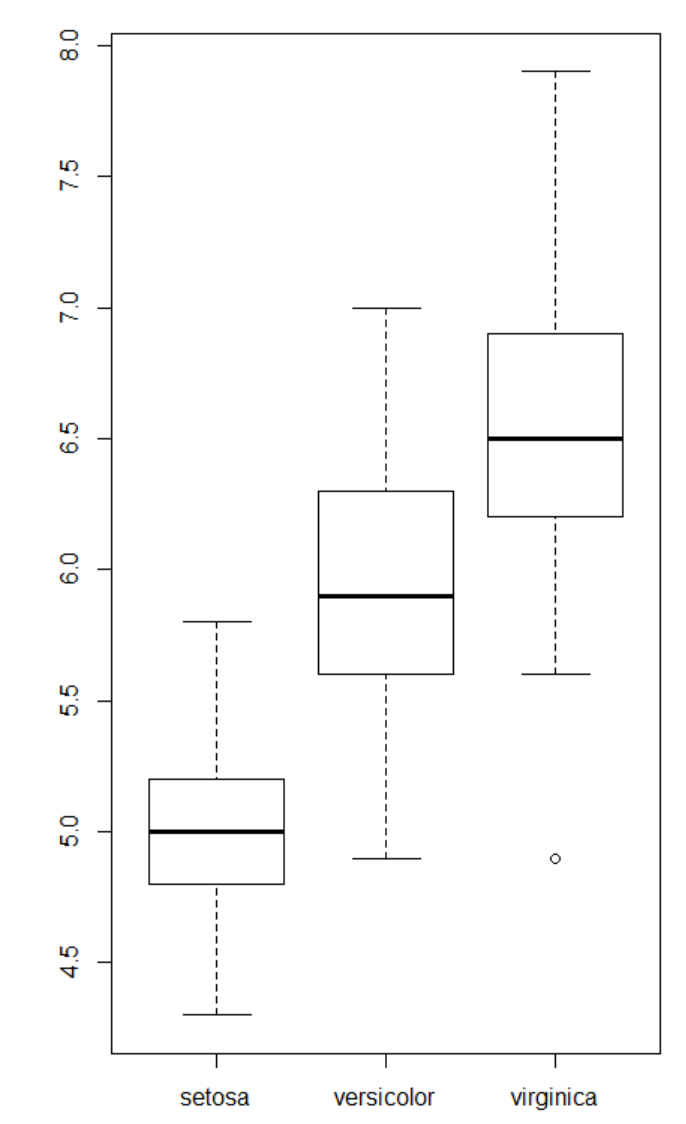
## Explore the ggplot2 package

Now that we have the data loaded, package loaded and we’ve done a small little dive at what is going on with the data, now we can look at the different types of charts ggplot2 will help create with the data.

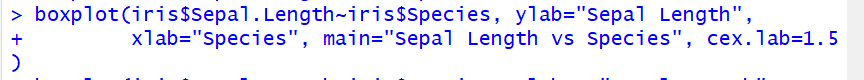
### Boxplot

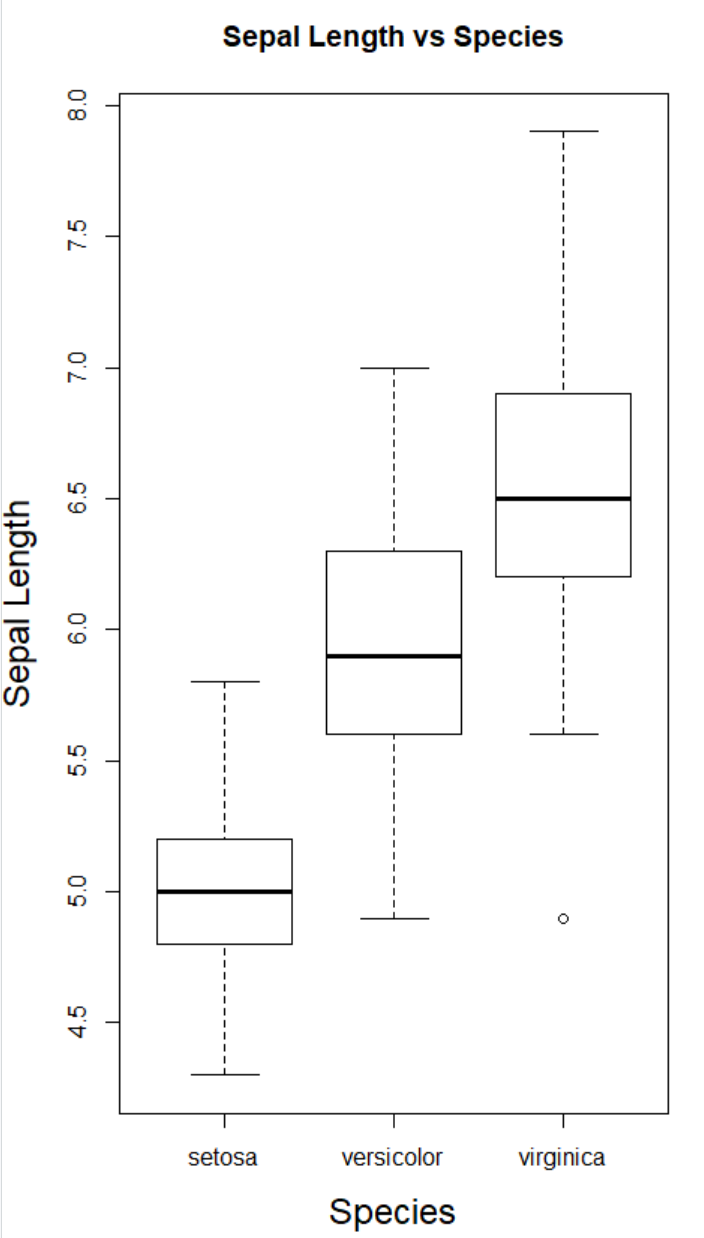
Boxplots are great for showing the data points which are deemed as outliers, also is good for finding where the median lies in the data spread, the difference in classification, as well as the quantile range within the distribution.



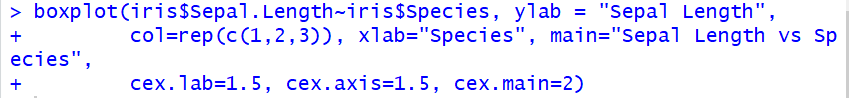


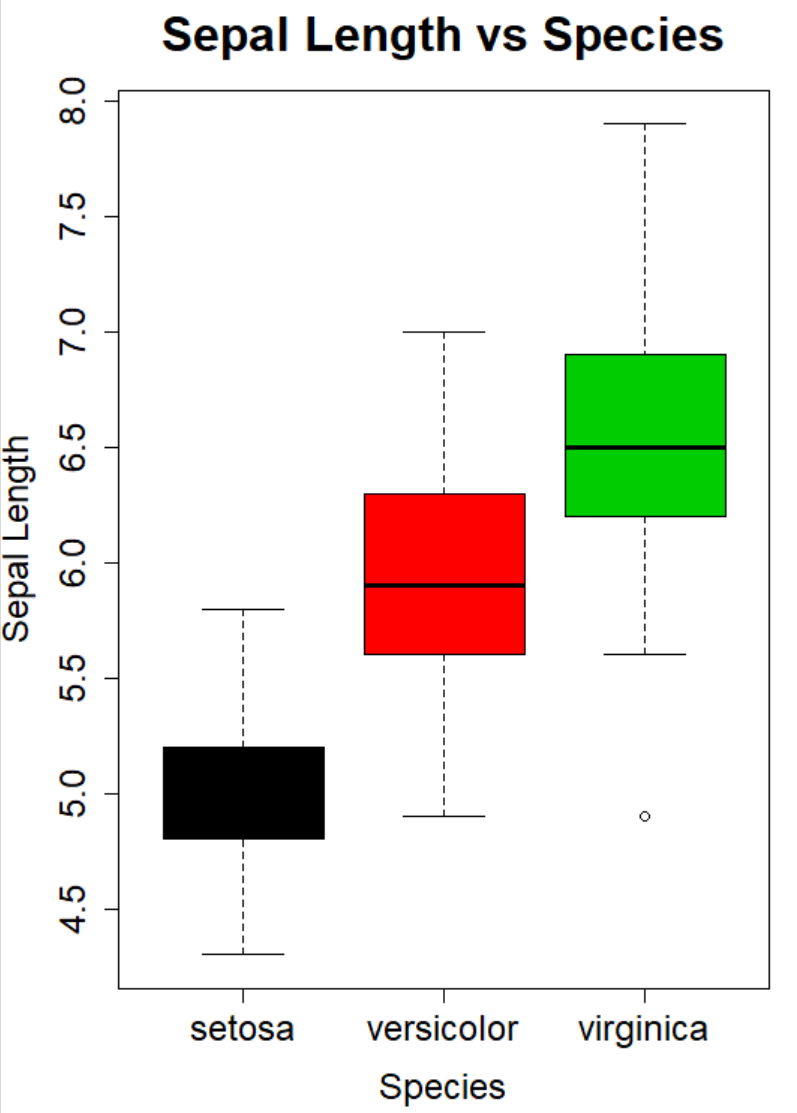
Boxplots are a standard plot within R. Above we have the distribution of Sepal Length vs. Species. Now let’s add some labels.





Then finally some color for each species.



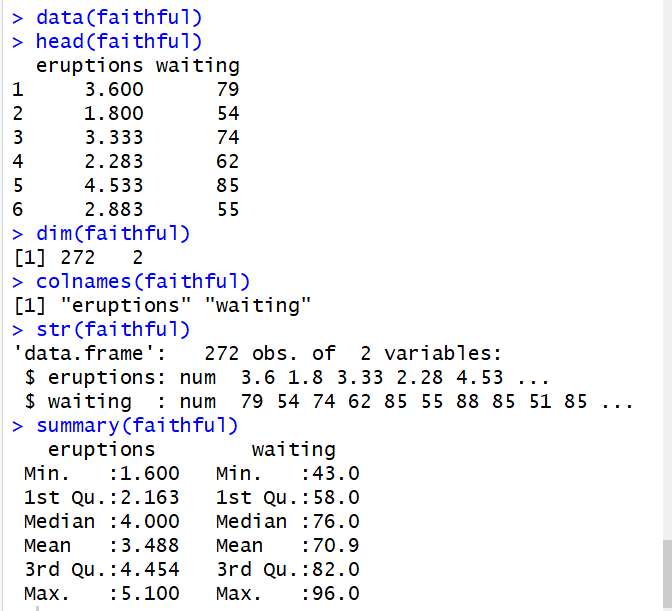


The color and labels add an extra layer to being able to analyze this data, it allows us to see what to expect from the sepal length of each of the species and how they vary because of it.

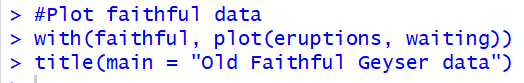
Now let’s look at the faithful dataset and do a little bit of analysis on it.

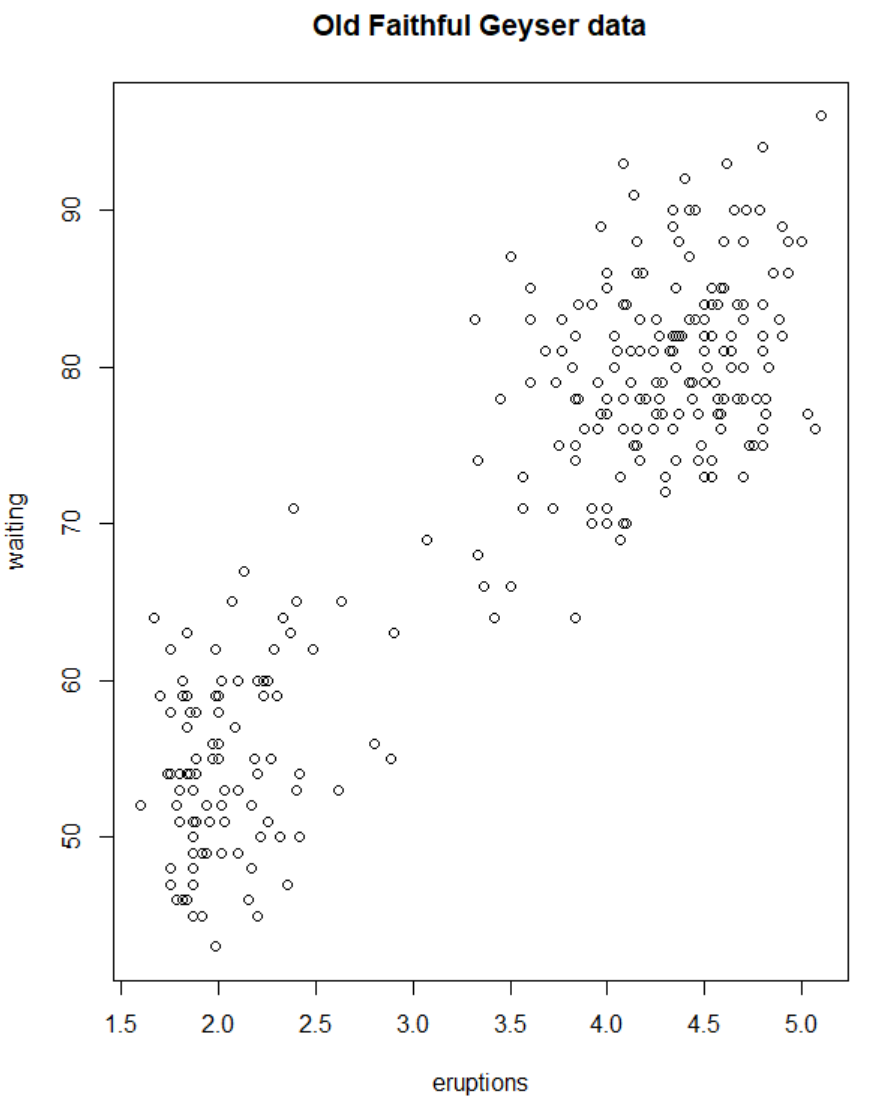
## Faithful Analysis

The faithful dataset is preloaded into RStudio about the geyser in Yellowstone national park, let’s take a quick look at the data before we create visualization.



Looking at the dataset, we have a dataset with only 2 variables but 272 records. The variables are the eruptions and the waiting time between eruptions. Both variables are numeric so now let’s plot the data.



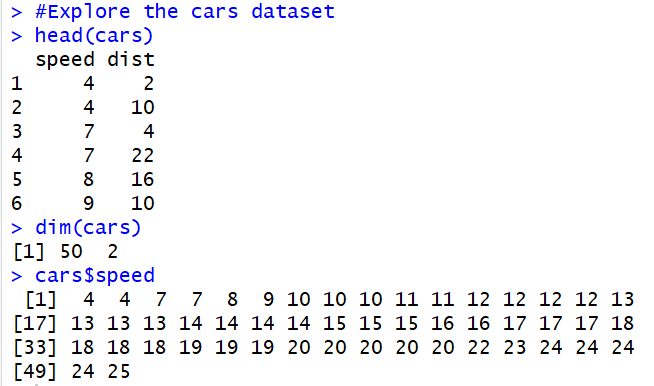


Plotting the data shows that the longer the wait time, the bigger the eruption appears to be the trend we can see in the data. A deeper dive into this data can make this more than just an assumption though.

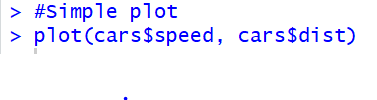
Next, let’s take a quick look at the cars dataset.

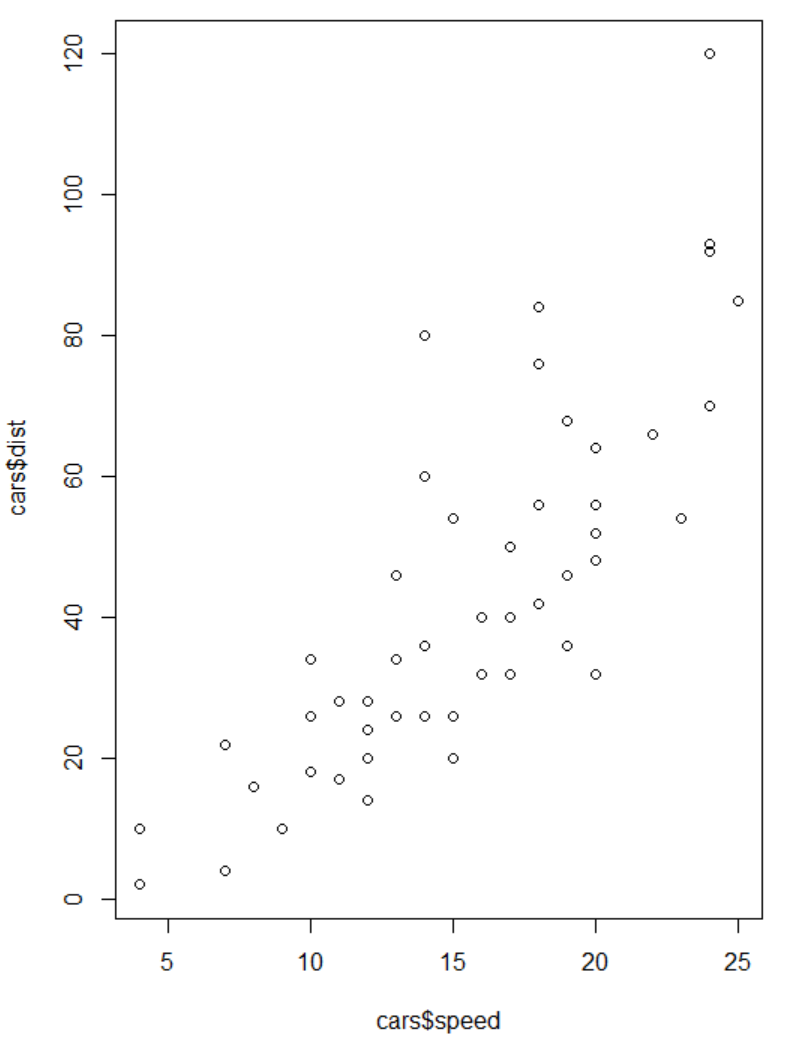
## Cars Analysis

Just like the faithful dataset, the cars dataset is a preloaded dataset in RStudio, let’s take a quick look at it before visualizing the data.

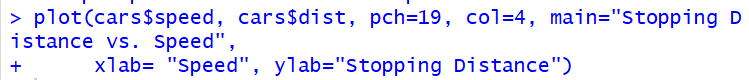


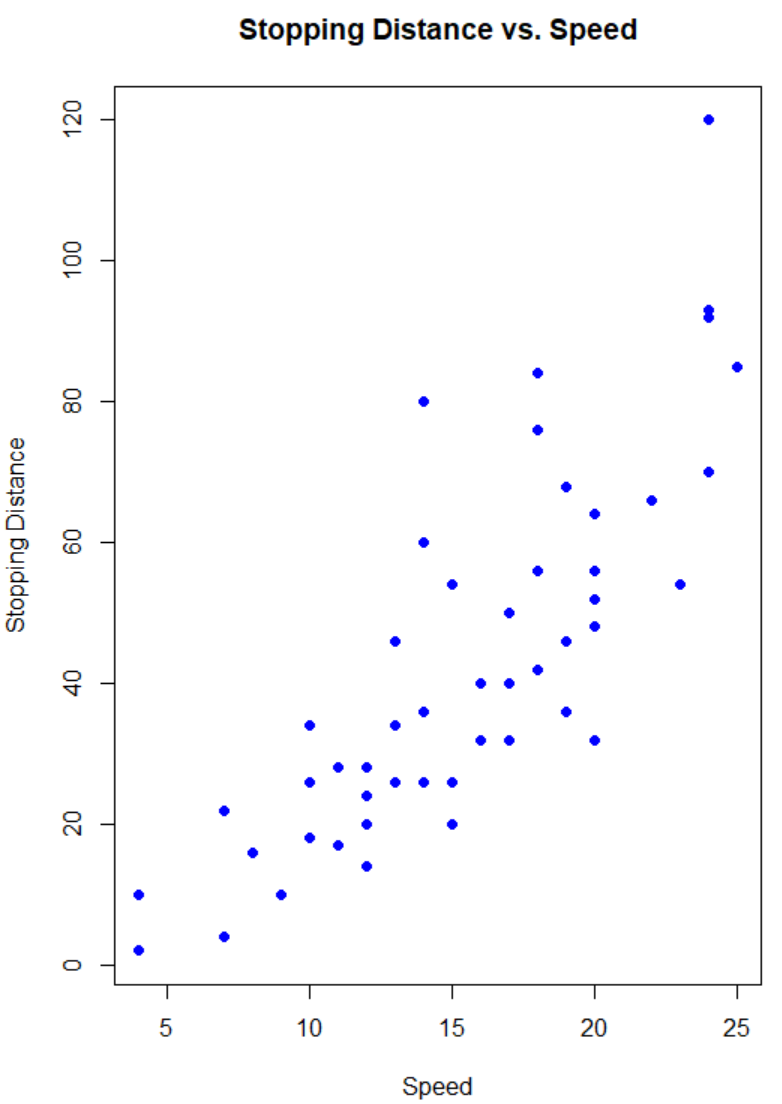
The dataset is pretty small, only 50 records and 2 variables. Both variables are numeric. Now let’s start plotting the data and seeing what we can uncover.

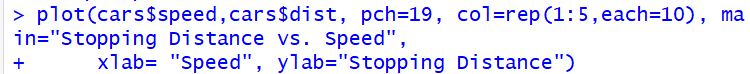


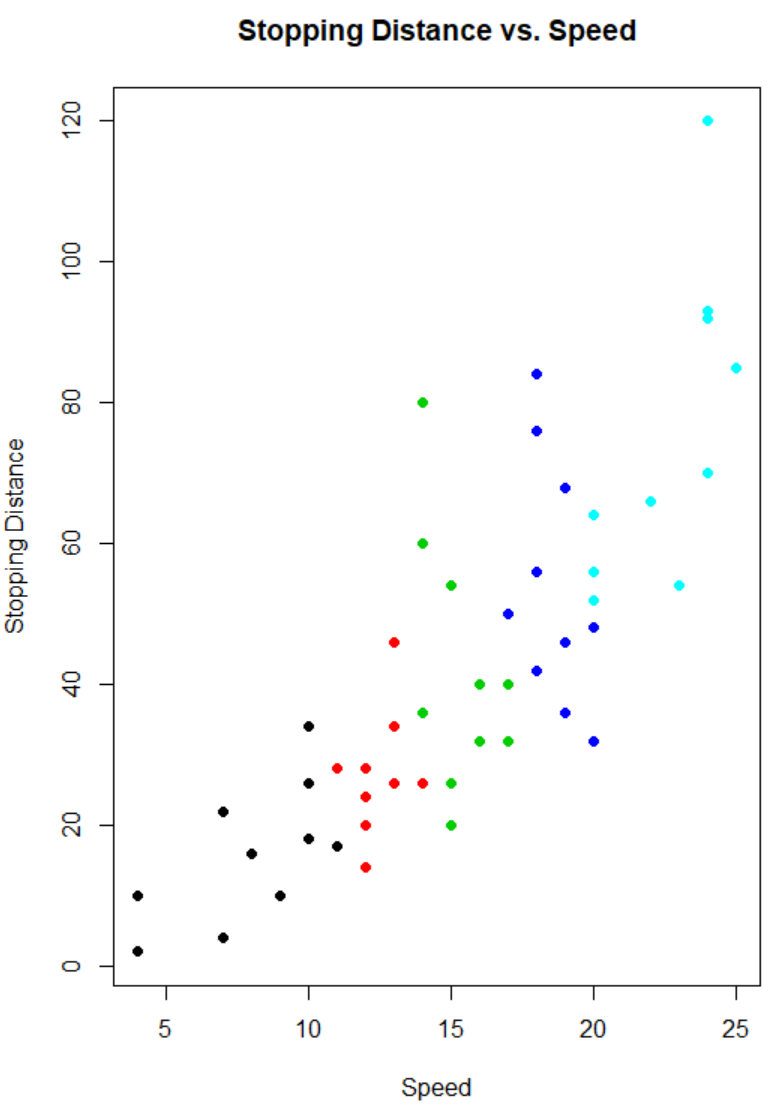


Now let’s color some data points.



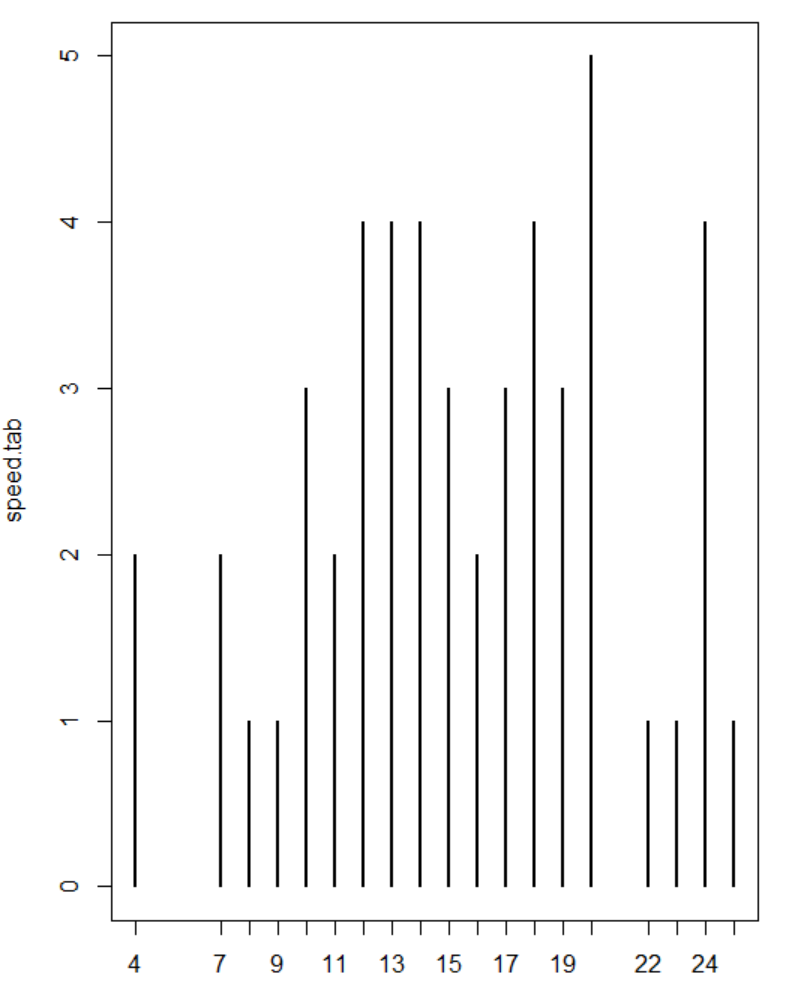






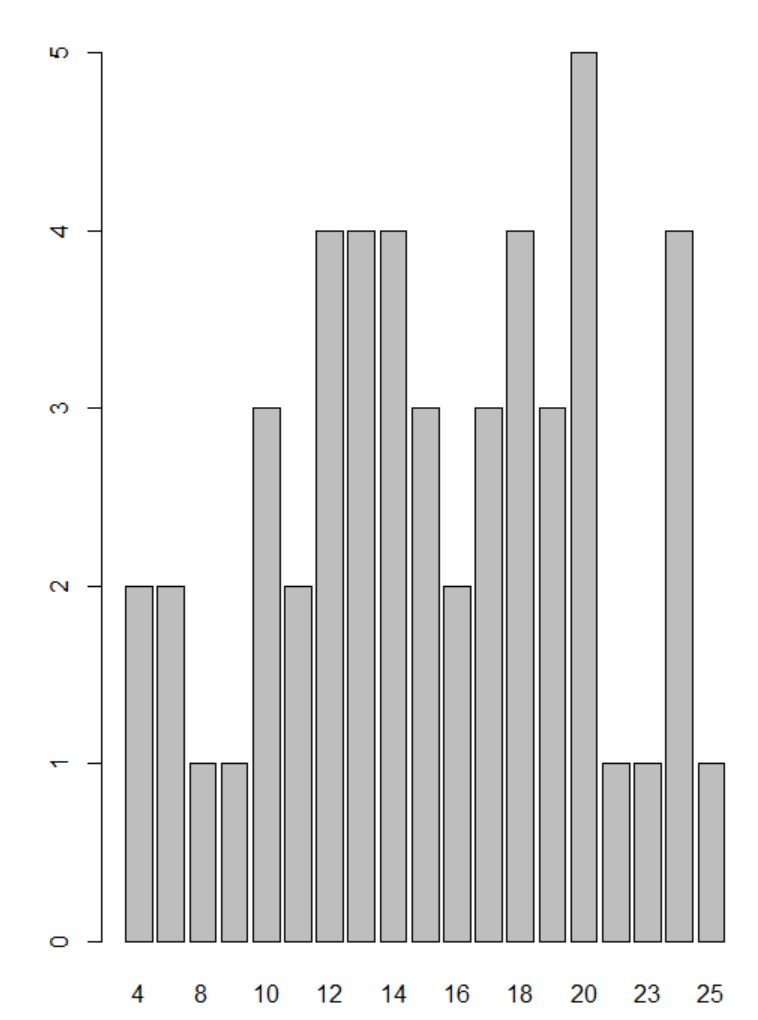
Now let’s table the speed column for this dataset, as well as plot this data.



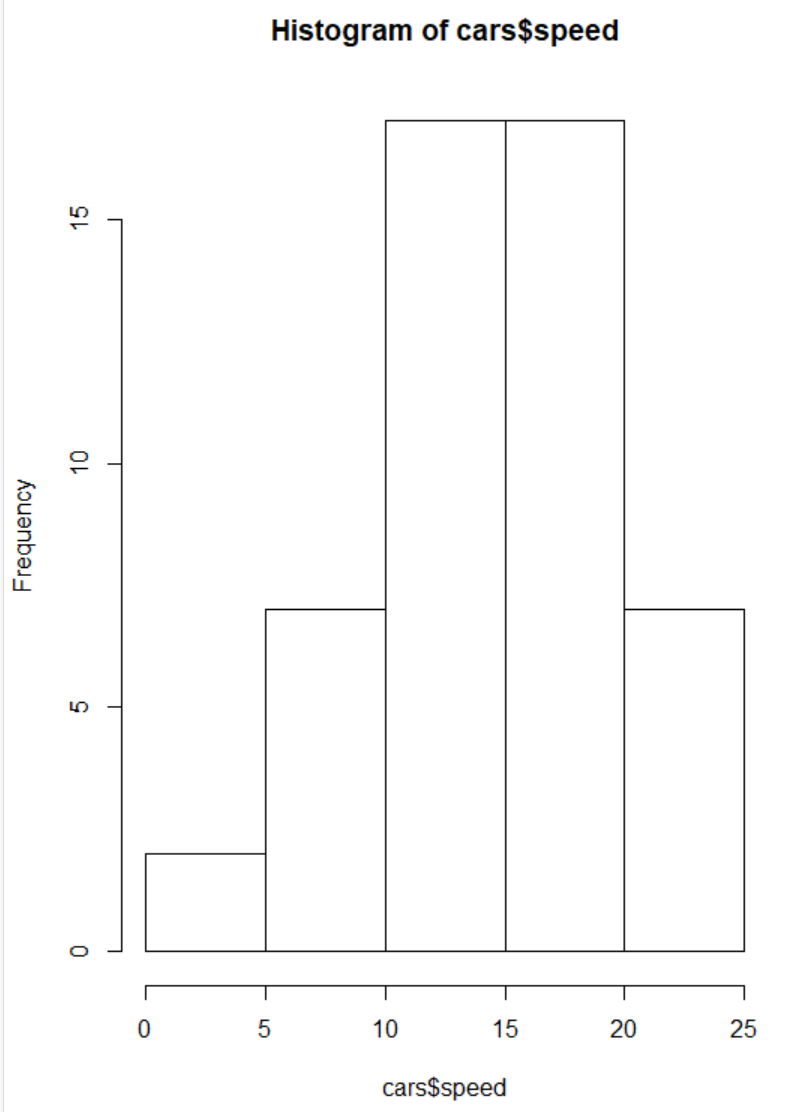


Interesting look for a plot, let’s look at some which are more effective at giving us information from the data.





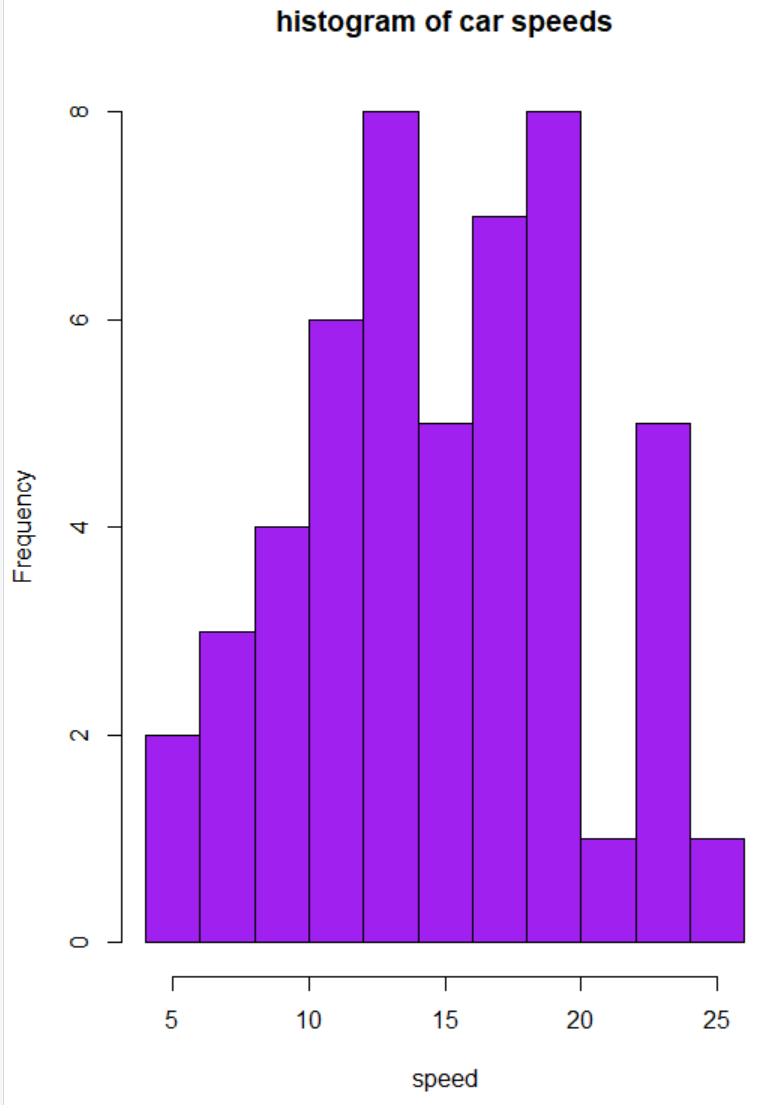




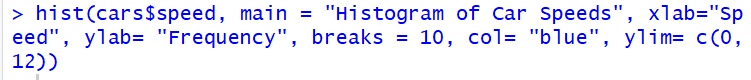
The bar plot does a good job of accurately showing the volumes based on the car speeds, then the histogram is good to put these car speed values into bins which can show which area of distribution do the speeds of most cars fall into.

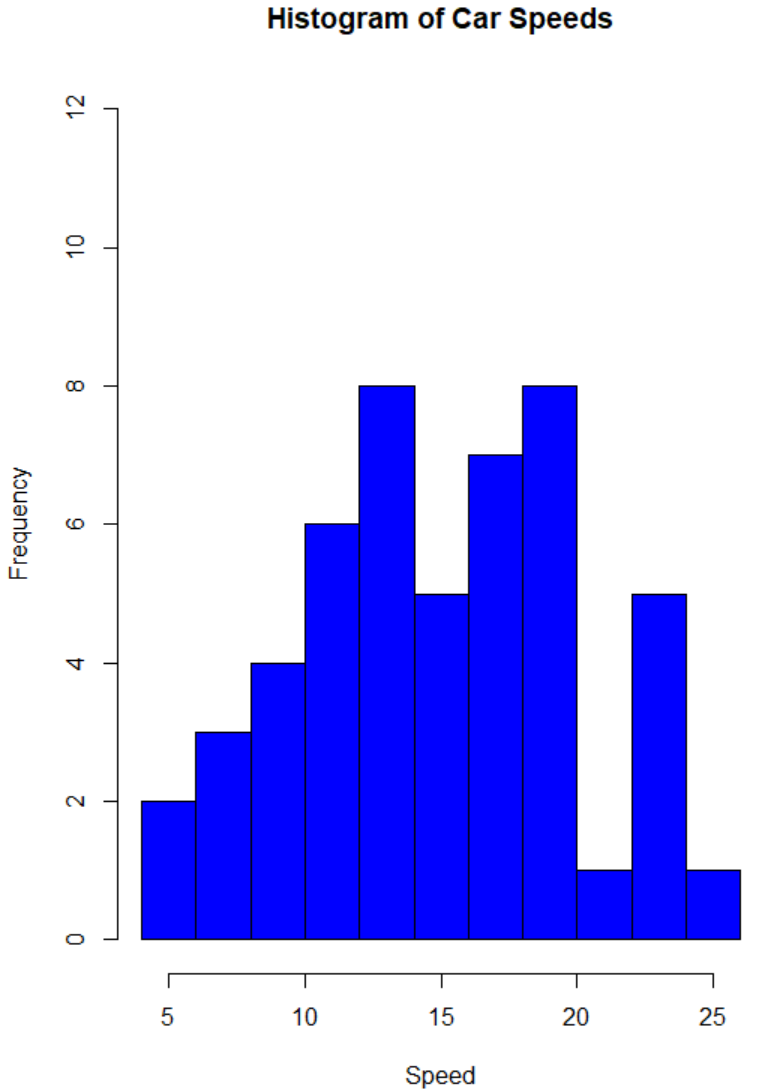
Let’s finish the histogram with a little bit more detail.





It appears our y-axis stops at 8, let’s stretch that limit to 12 so we give it more space.

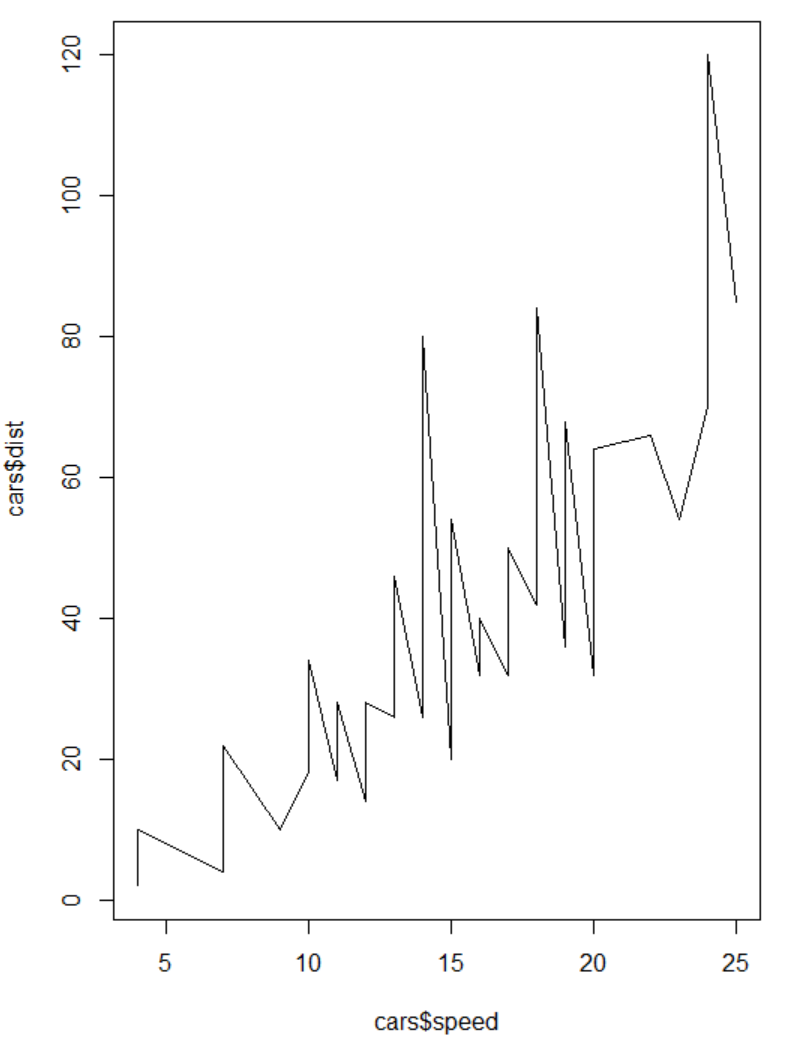




That is a good distribution with the a good amount of spacing on the y-axis to clearly show the frequency of the bins has a maximum of 8.

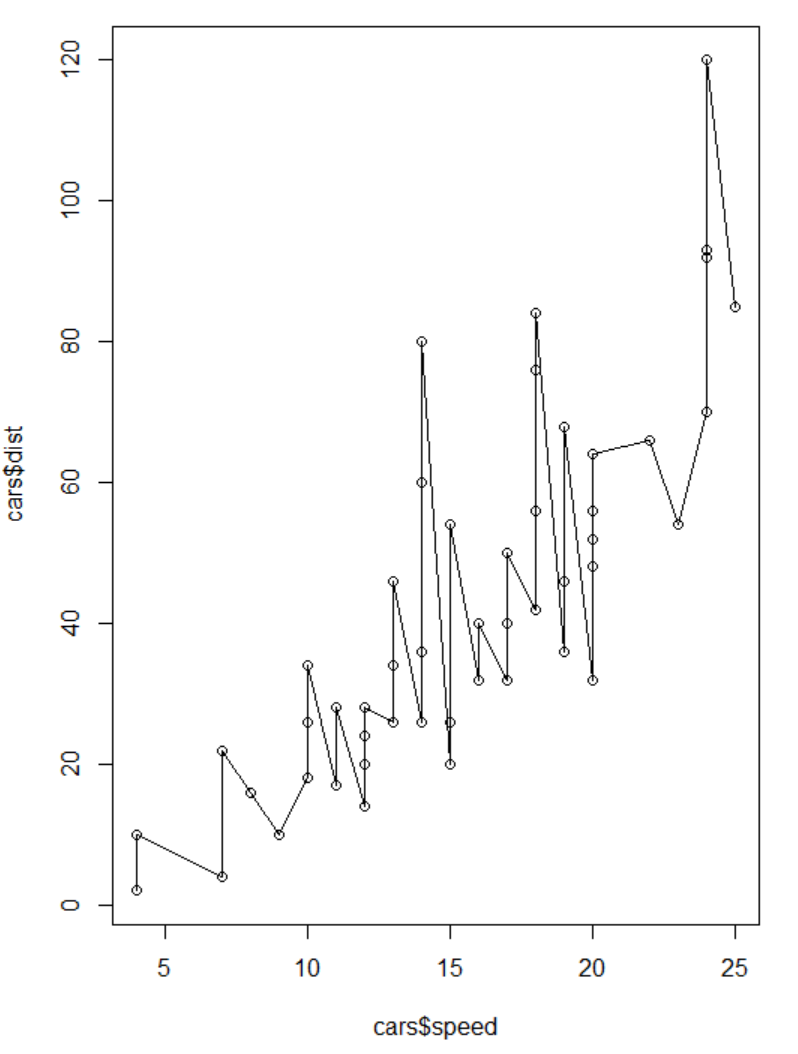
Now let’s look at a regression plot of the cars data and uncover some more useful information about this dataset.



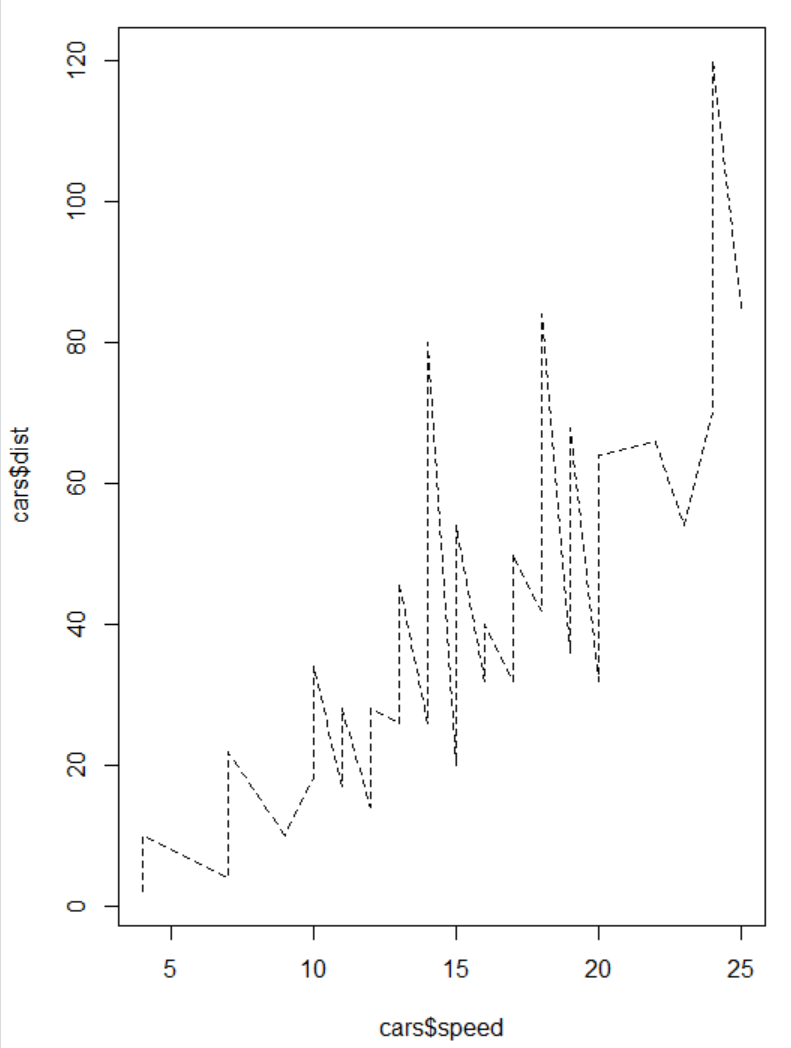


You can change how the line is presented within the chart. For example I can change the type to do a better job of showing the data point, or I can make the line a dashed line.



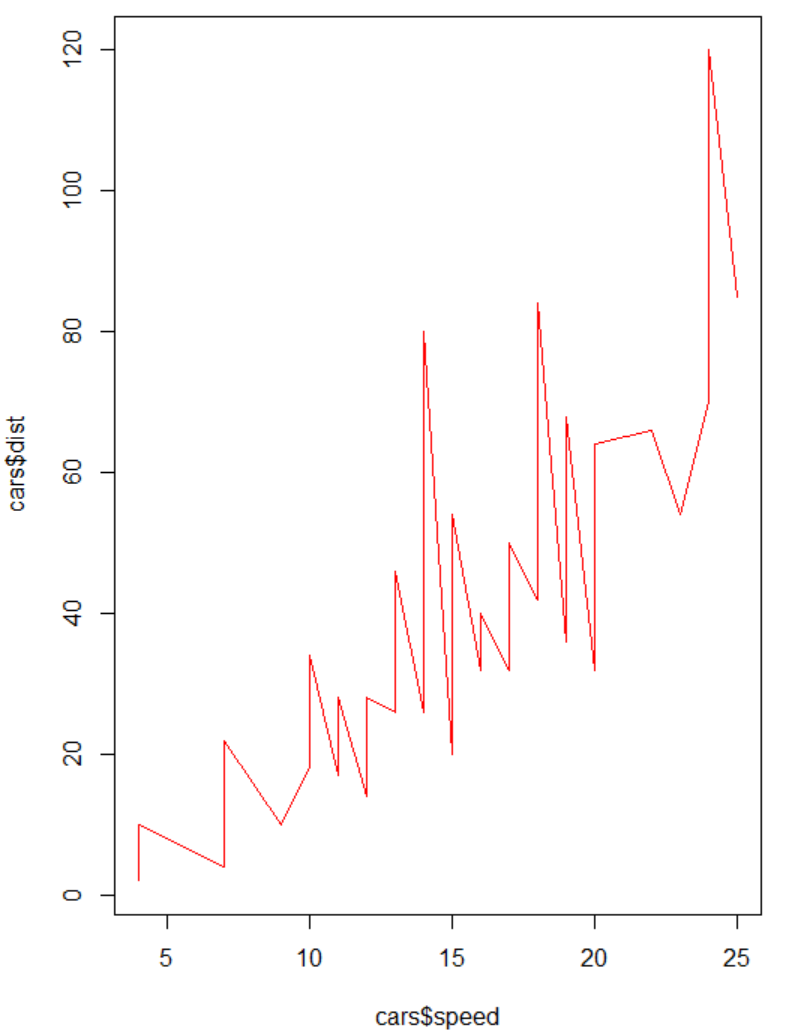






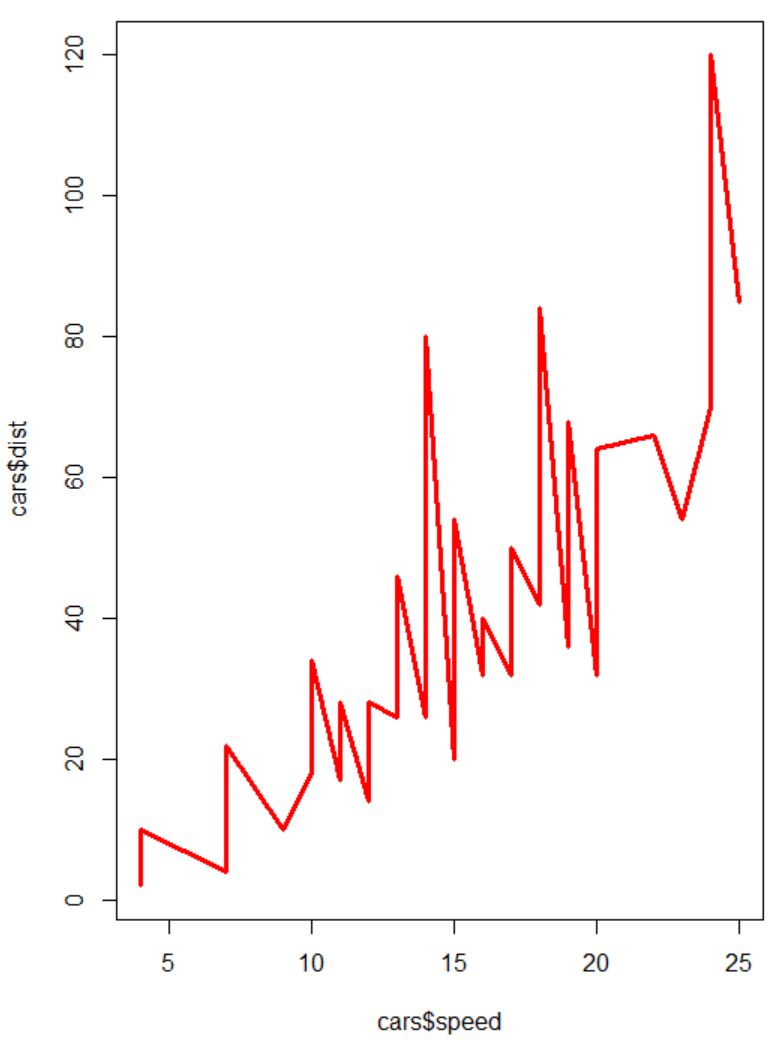
There are several different options for how a line can be displayed in RStudio depending on your need. I would like to also add some color to my chart so I will do that next.



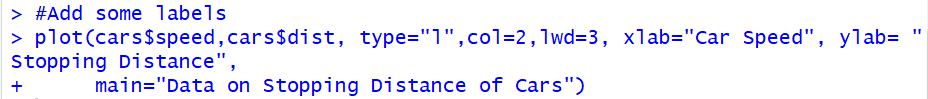


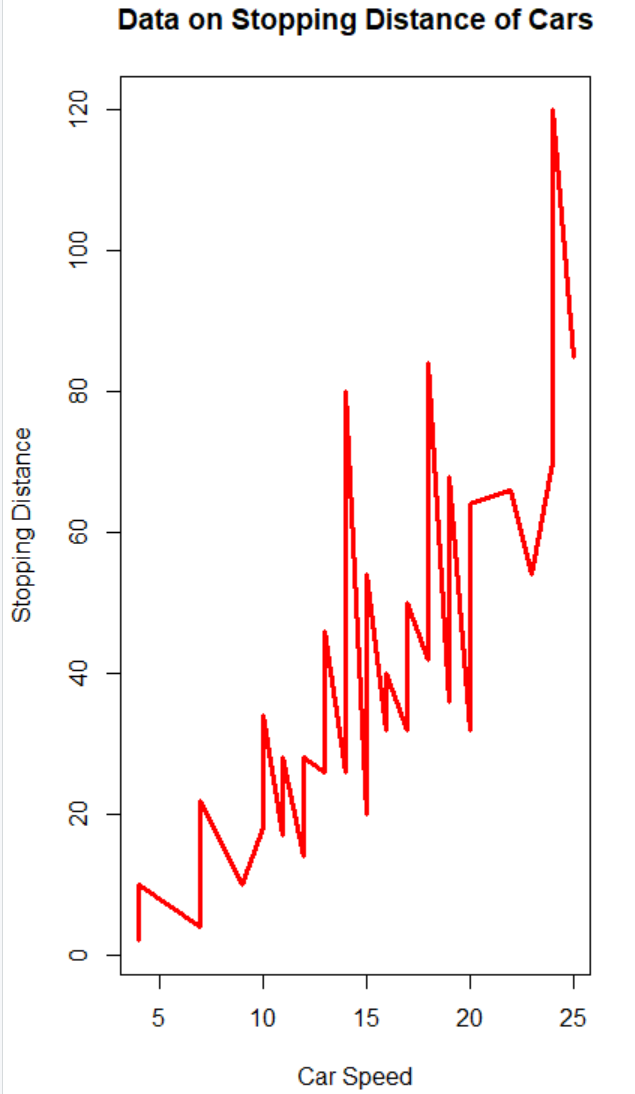
What this shows here, is that in RStudio, you can change the color of a chart either by putting a specific color in “ “ or by an assigned number. We can also change the thickness of the line as well.



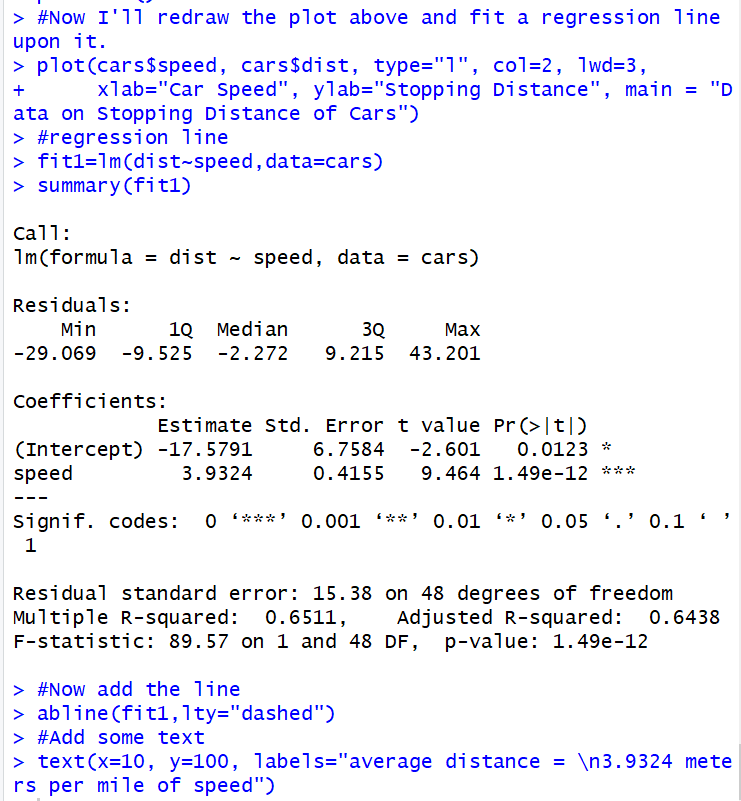


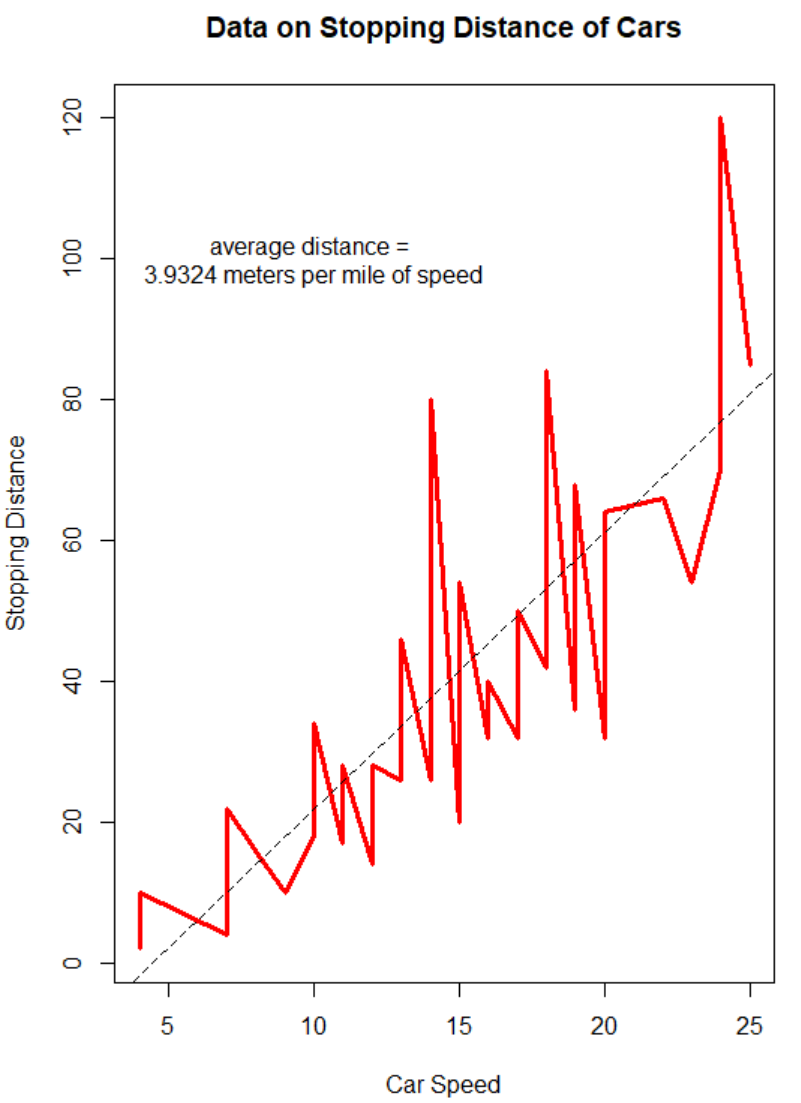
Now let’s add some labels.





Next, I’ll wipe it clear with a plot.new(), then redraw the plot with the regression line upon it.





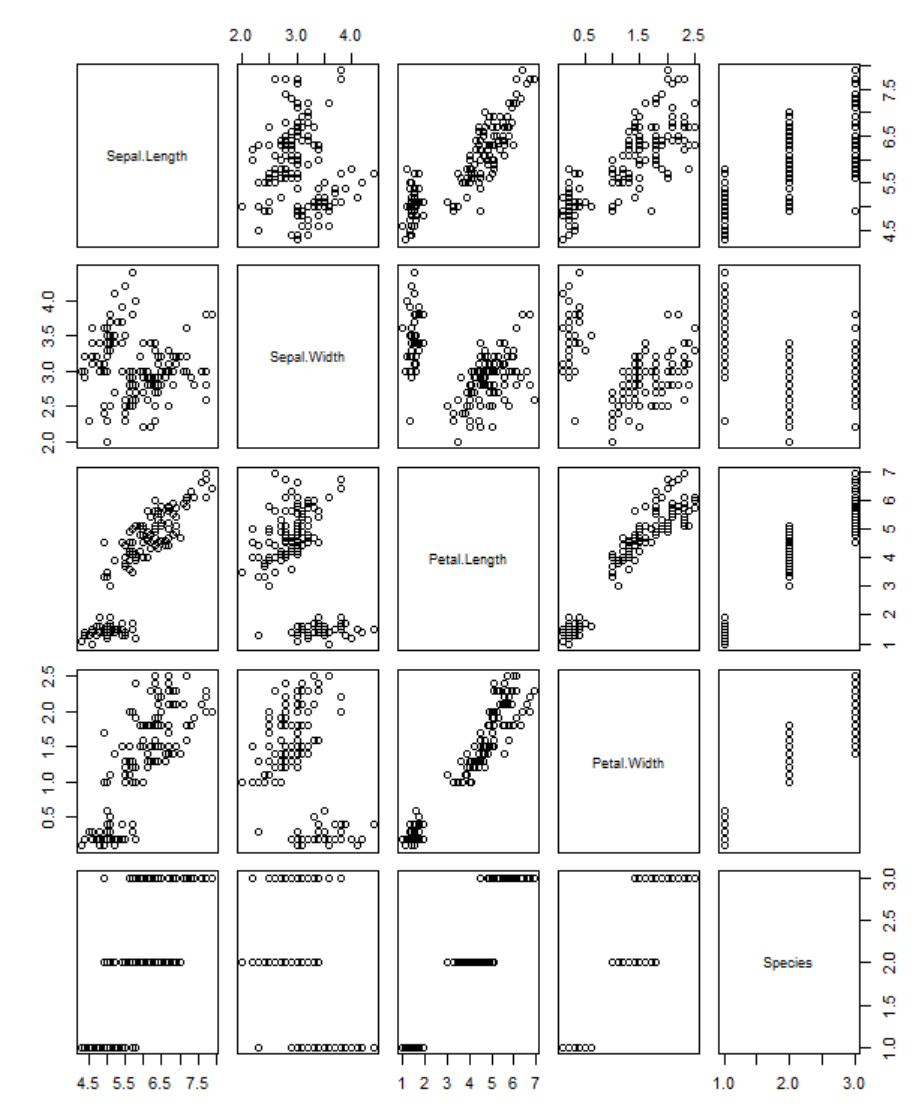
As seen above, the cars dataset is going about 3.9 meters per mile of speed for its average distance.

Now, I’ll go back to the iris dataset and put together a scatterplot which institutes the differences in the three species and shows how they differ.

## Iris Dataset and building a Scatterplot

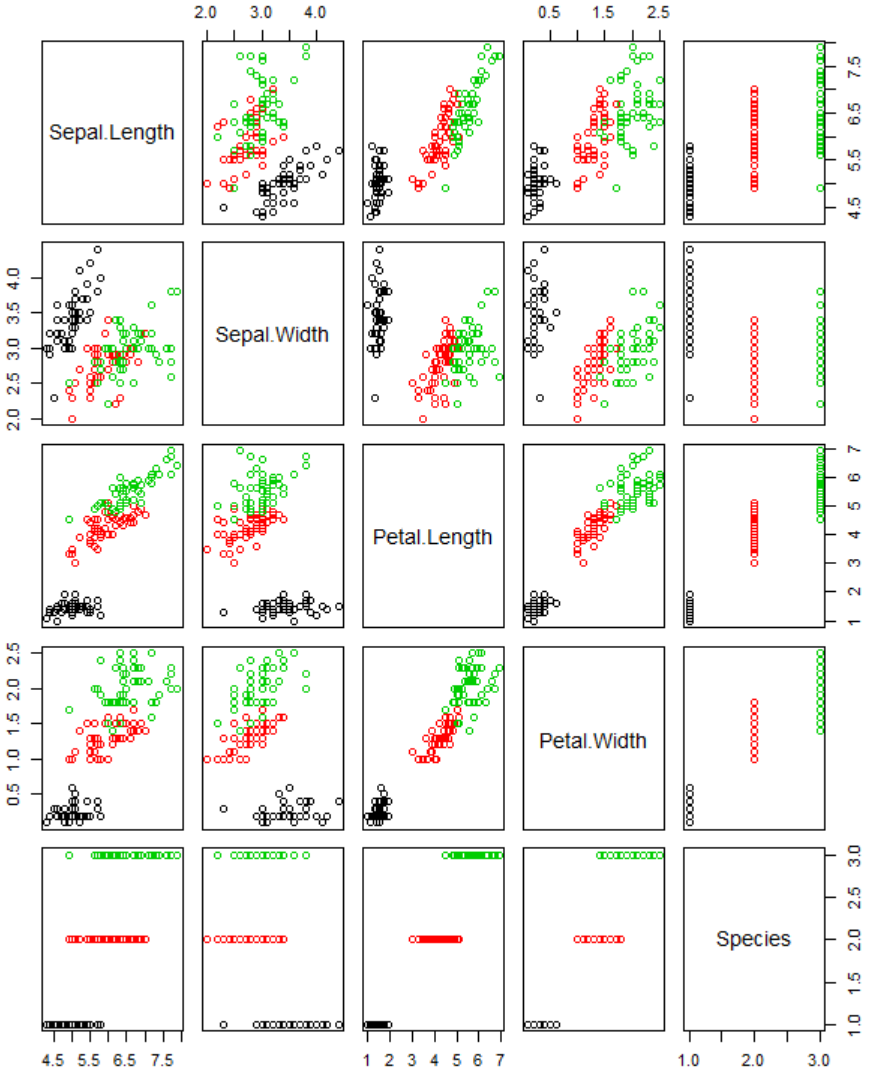
Let’s start by looking at the scatter plot pairs of the variables within the Iris dataset.





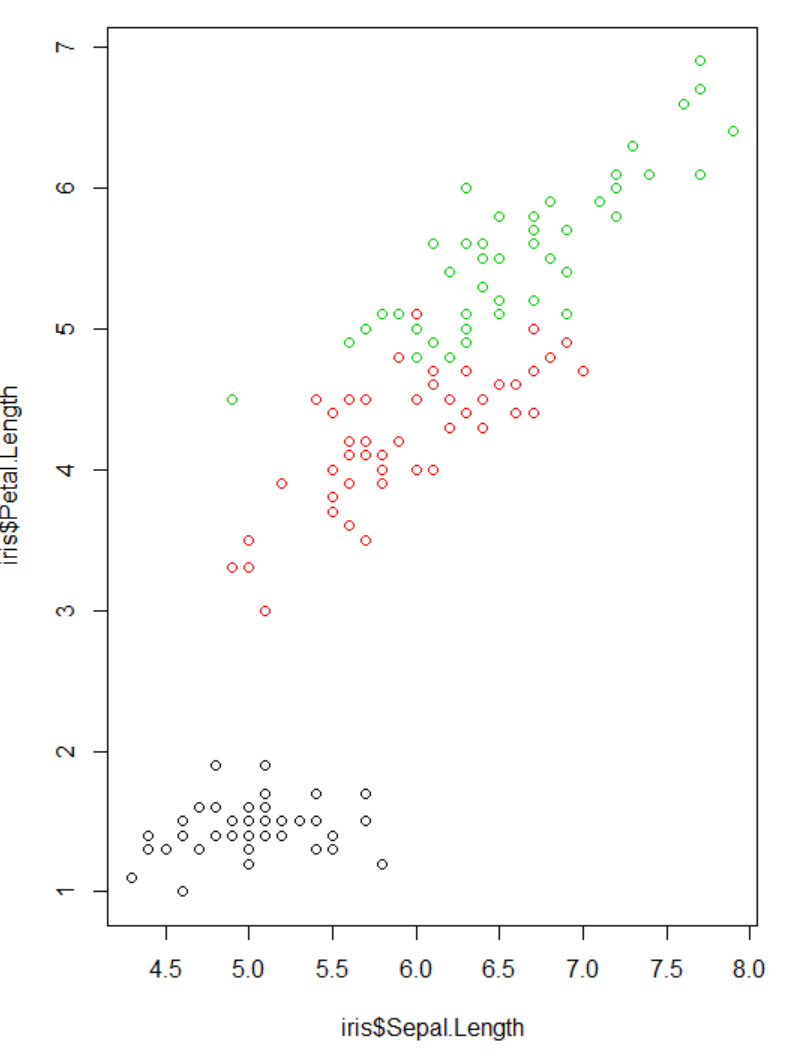
The pairs show the possible correlations between variables. I find it interesting looking at any of the variables matched with Species and how there is always 3 distinct groups in all 8 of those charts. It is easier to see this trend when we add some color to the evaluation.





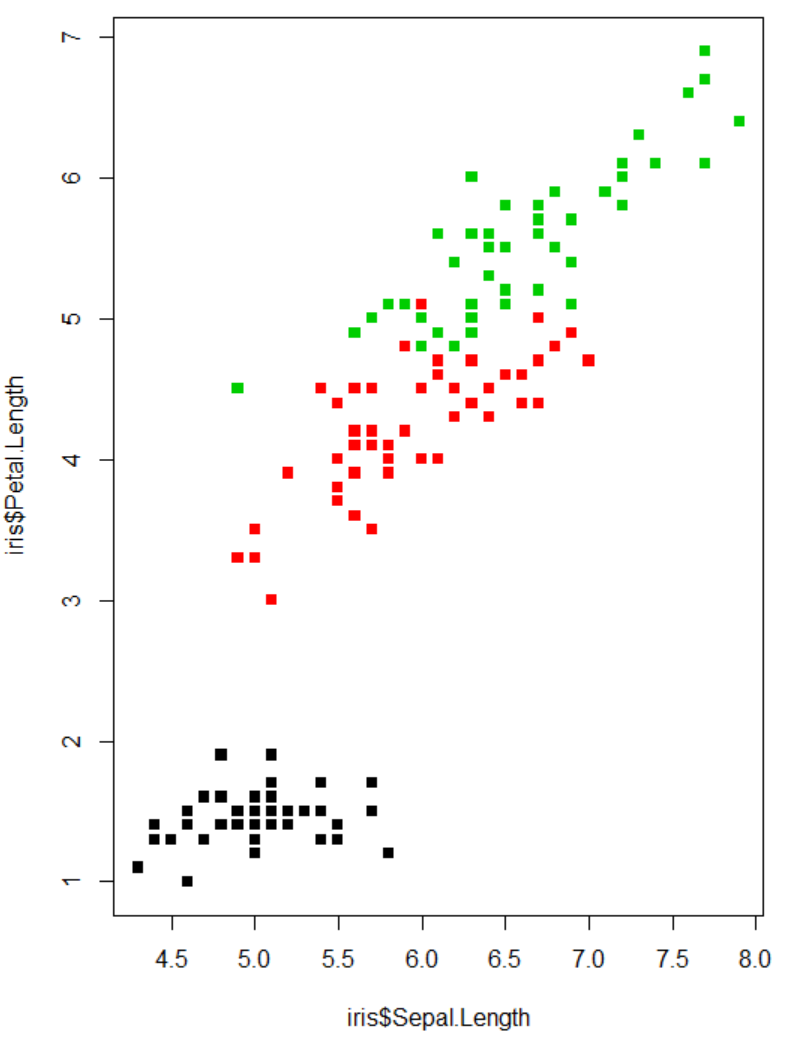
Now let’s look at the petal length and sepal length within its own scatterplot, we will also color the data points differently based on their species.



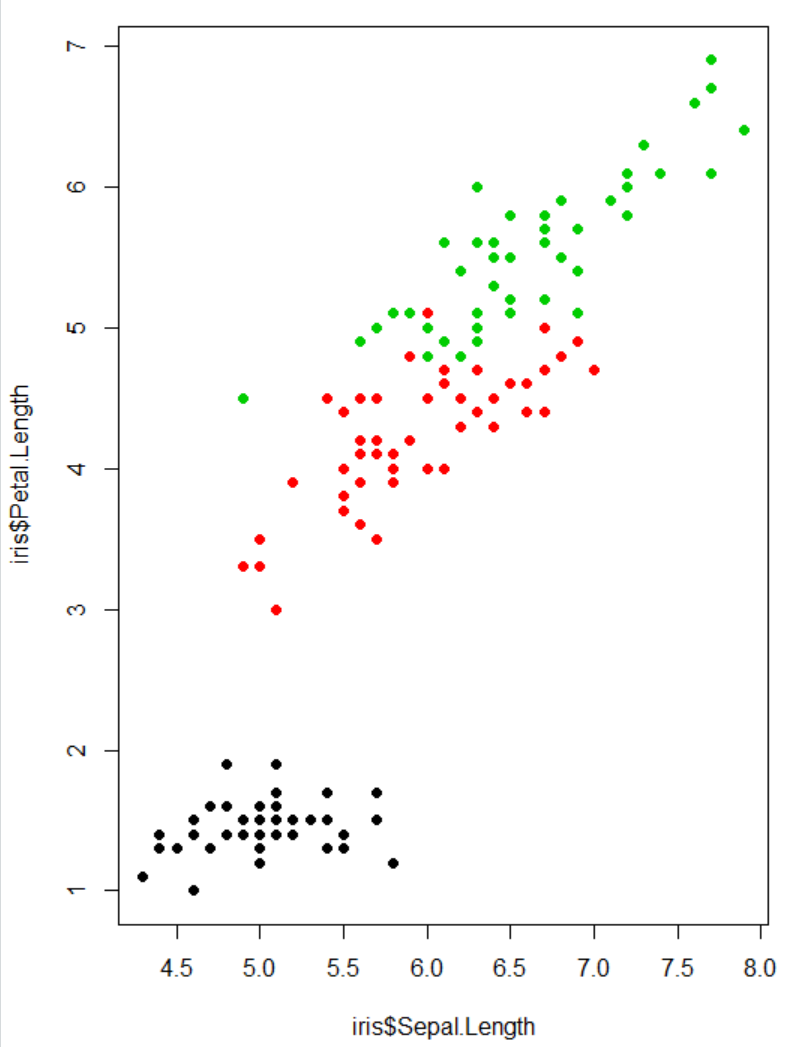


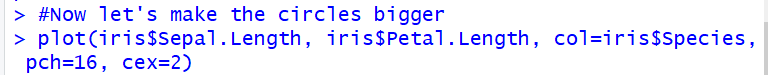
We can also mess around with the data points and try and out different shapes as well as fill them in as well. Check out these scatterplots.

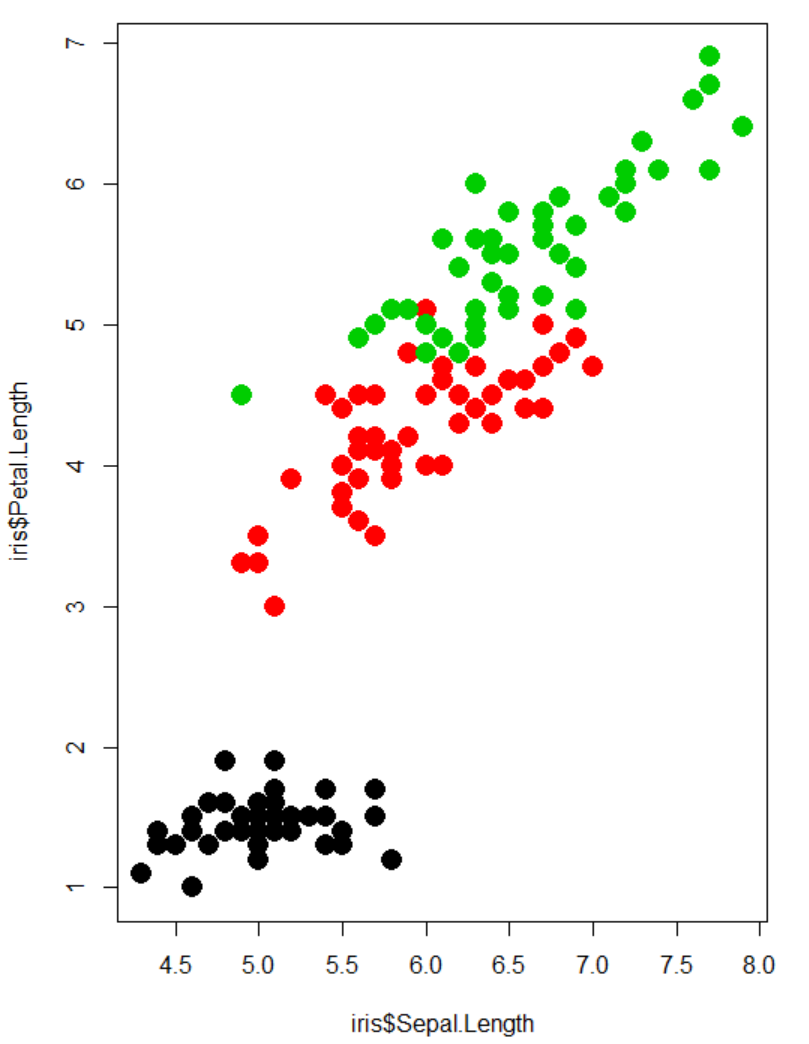




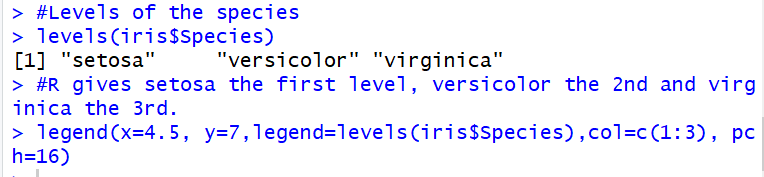


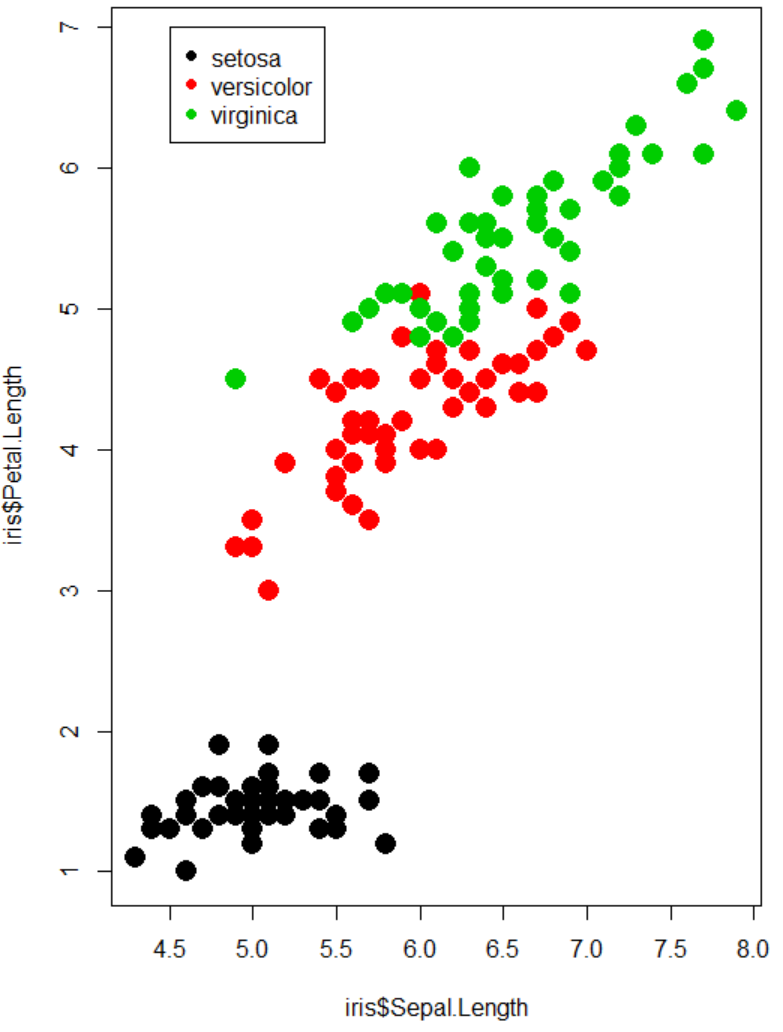






Now that we’ve made the scatterplot with the data points colored by species let’s make a legend so anyone who is reading this chart can know the difference.

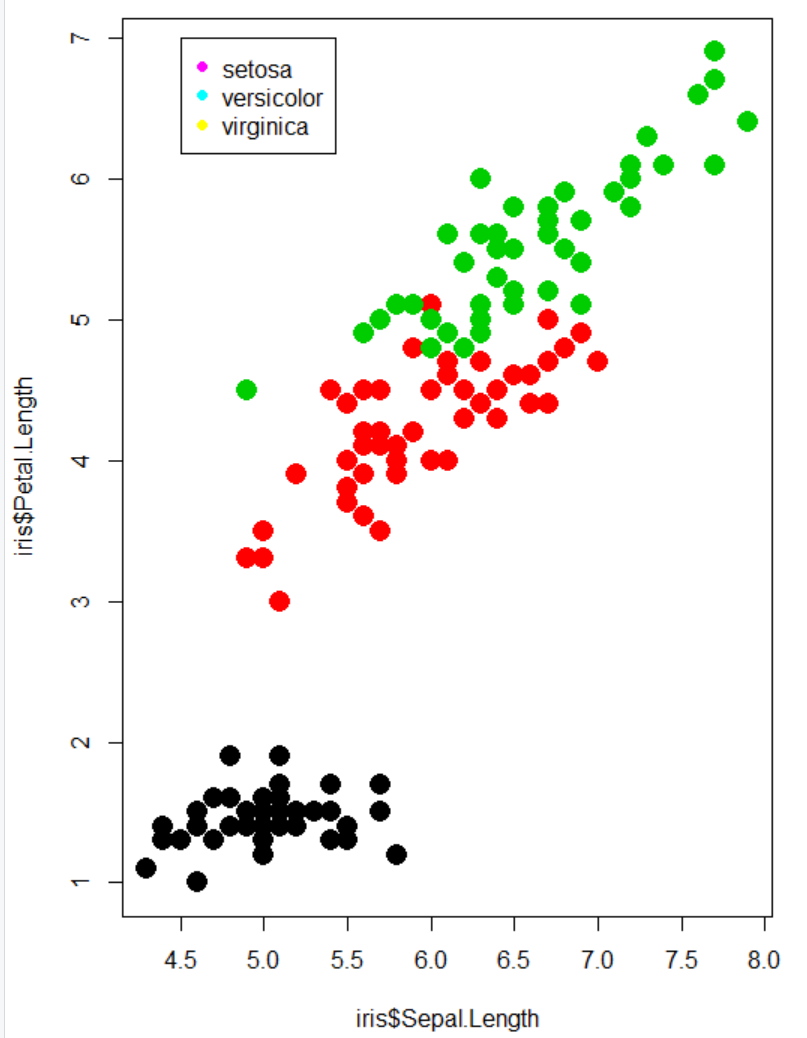




I started by looking at the levels of the Species column to see the names of each species, then I added the levels as the legend in the chart.

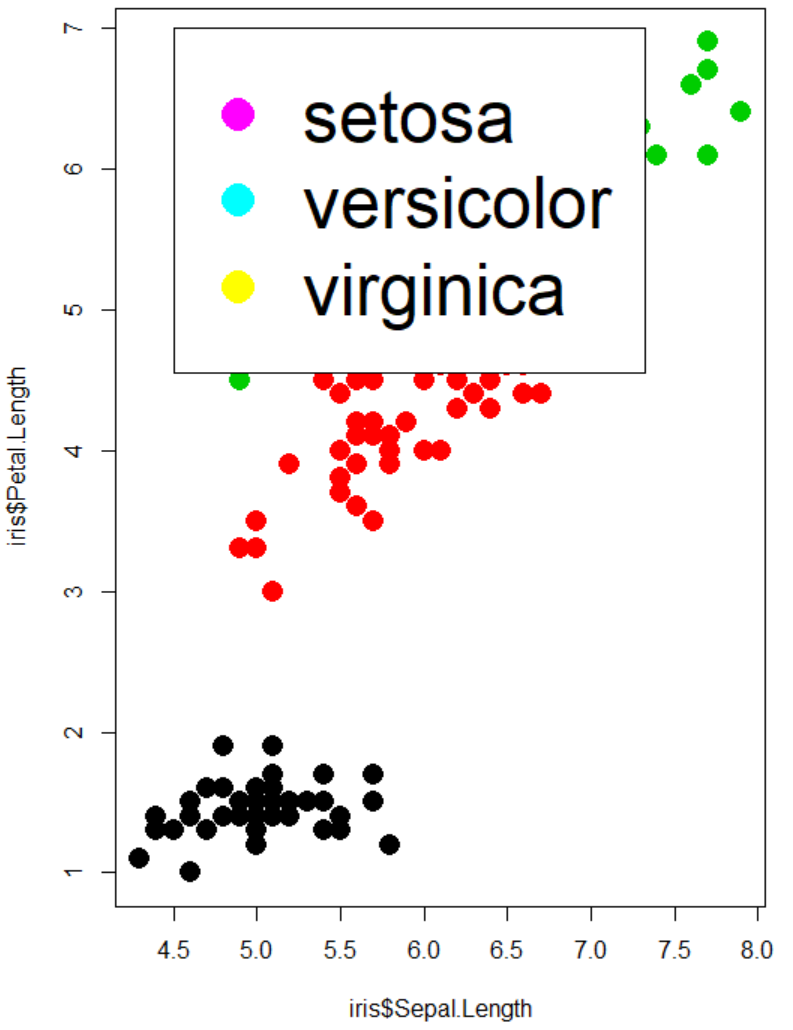
To match what we have at the beginning of the From the Expert text, we need to change the colors in the legend, this is all we need to do to accomplish that.





And this is what it looks like with excessive size!





When the legend isn’t so big that it blocks out the data we have a chart which does a good job showing the data points sepal length to their petal length as well as show how different each of the three species really are by those two variables.

## Reflection

Data visualizations are a very important part of exploring a dataset. It is a great way to show the trends in the data in a way a normal data table or some summary statistics just can’t accomplish. What makes the ggplot2 package great for any analysis done in RStudio is that it really helps highlight those differences in the data which are useful information and what R itself just can’t provide.