Introduction to EDA with R

== R and Data Types == - Strangely Typed Language - Coercible Types - Everything has 3 effective types: type, storage mode, class

```
x <- 5
42.01 -> y
v <- c(1,2,3,4,5)
typeof(x)
```

```
## [1] "double"
```

```
typeof(y)
```

```
## [1] "double"
```

```
typeof(v)
```

```
## [1] "double"
```

```
class(x)
```

```
## [1] "numeric"
```

```
class(y)
```

```
## [1] "numeric"
```

```
class(v)
```

```
## [1] "numeric"
```

```
mode(x)
```

```
## [1] "numeric"
```

```
mode(y)
```

```
## [1] "numeric"
```

```
mode(v)
```

```
## [1] "numeric"
```

```
x
```

```
## [1] 5
```

```
y
```

```
## [1] 42.01
```

```
v
```

```
## [1] 1 2 3 4 5
```

- Dimension

```
dim(v)
```

```
## NULL
```

```
length(v)
```

```
## [1] 5
```

```
length(x)
```

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

- indexing starts at 1

```
x[1]
```

```
## [1] 5
```

```
y[1]
```

```
## [1] 42.01
```

```
v[1]
```

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

- In R, vectors are the fundamental type

```
x * v
```

```
## [1]  5 10 15 20 25
```

- Heterogeneous Types

```
v2 <- c(1, 2, 3, 4, "Hello")
typeof(v2)
```

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

```
mode(v2)
```

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

```
class(v2)
```

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

```
v2
```

```
## [1] "1"     "2"     "3"     "4"     "Hello"
```

- type conversion as.

```
7 * as.numeric(v2[1:4])
```

```
## [1]  7 14 21 28
```

```
v2
```

```
## [1] "1"     "2"     "3"     "4"     "Hello"
```

== Data Frame == - Complex object - Parallel vectors in a structure

```
x <- c(1,2,3,4)
y <- rnorm(4, mean=38, sd=4)
z <- c(TRUE, FALSE, FALSE, TRUE)
df <- data.frame(x, y, z)
df
```

```
##   x        y     z
## 1 1 40.93501  TRUE
## 2 2 33.04471 FALSE
## 3 3 36.06598 FALSE
## 4 4 37.83711  TRUE
```

```
typeof(df)
```

```
## [1] "list"
```

```
mode(df)
```

```
## [1] "list"
```

```
class(df)
```

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

- Indexing a data frame

```
#columns
typeof(df[1])
```

```
## [1] "list"
```

```
typeof(df[2])
```

```
## [1] "list"
```

```
typeof(df[3])
```

```
## [1] "list"
```

- Refer to the attributes of a data frame

```
df$x
```

```
## [1] 1 2 3 4
```

```
typeof(df$x)
```

```
## [1] "double"
```

- add a column, base this on some kind of match

```
df$Class[df$x %% 2 == 0]  <- "Even"
df$Class[df$x %% 2 == 1]  <-"Odd"
```

```
df$x %% 2 == 0
```

```
## [1] FALSE  TRUE FALSE  TRUE
```

== Iris Exploratory Data Analysis ==

```
data(iris)
class(iris)
```

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

```
head(iris)
```

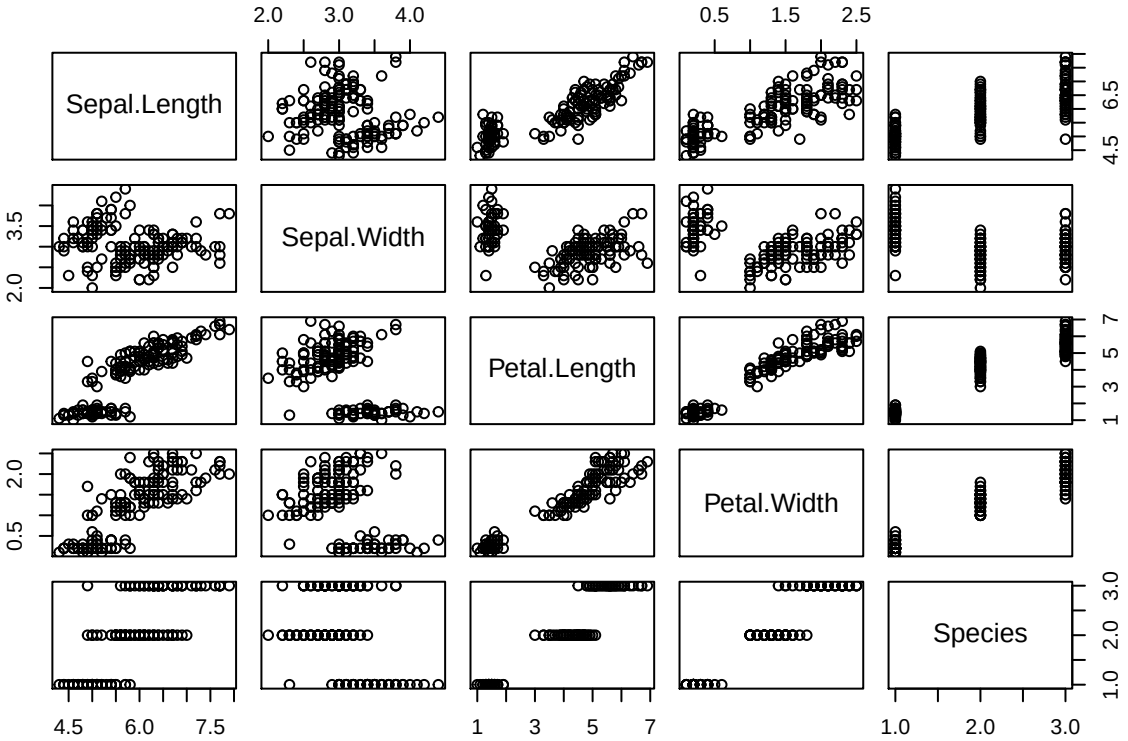
```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
```

```
## 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
```

```
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
##
##
##
```

```
plot(iris)
```



Create a color varible in the iris data frame

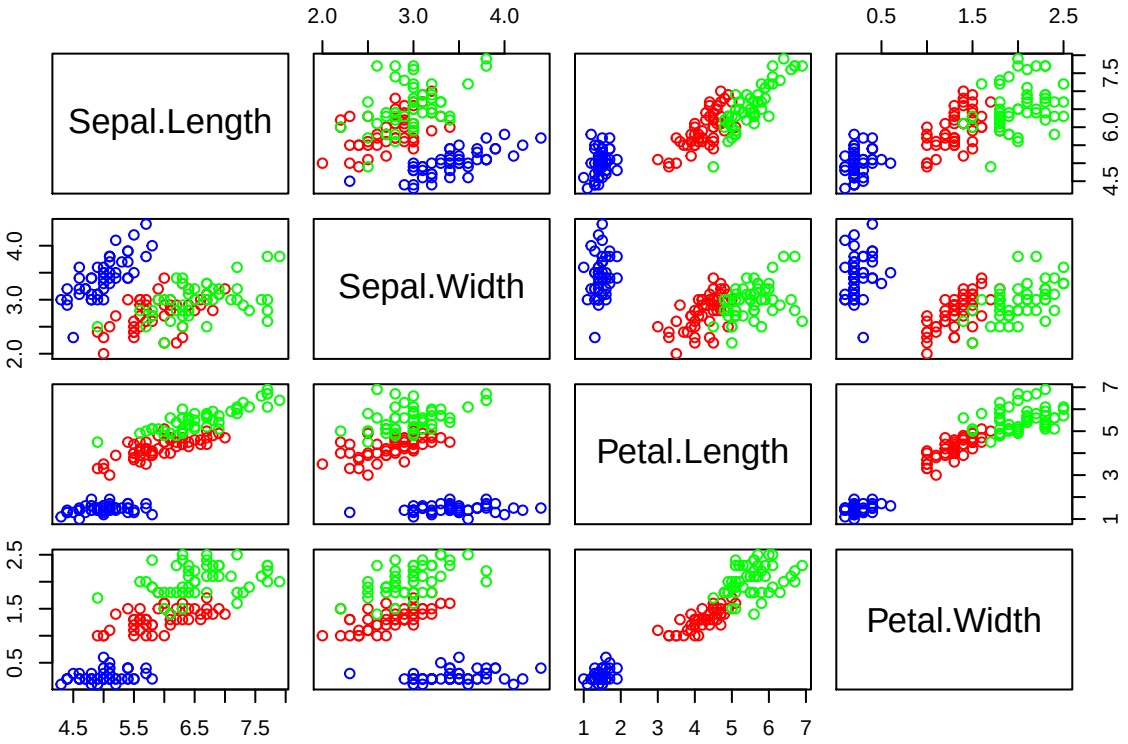
```
unique(iris$Species)
```

```
## [1] setosa     versicolor virginica
## Levels: setosa versicolor virginica
```

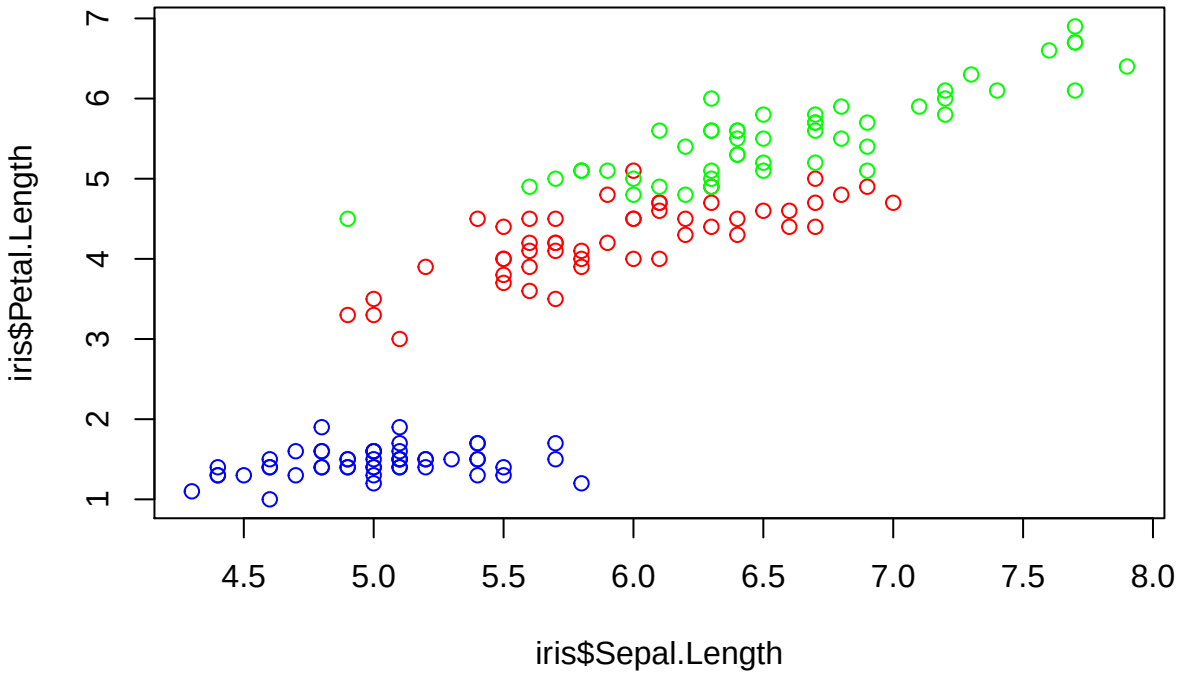
```
iris$Color[iris$Species == "setosa"] = "blue"
iris$Color[iris$Species == "versicolor"] = "red"
iris$Color[iris$Species == "virginica"] = "green"
```

plot the numeric variables with colors

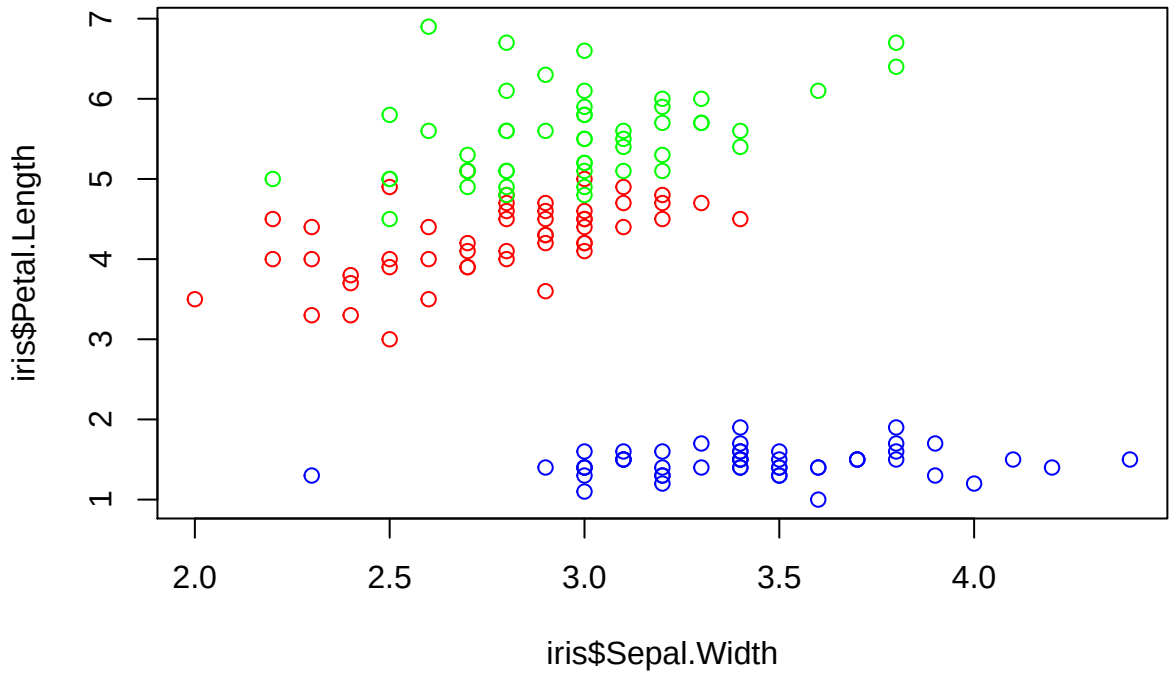
```
plot(iris[1:4], col=iris$Color)
```



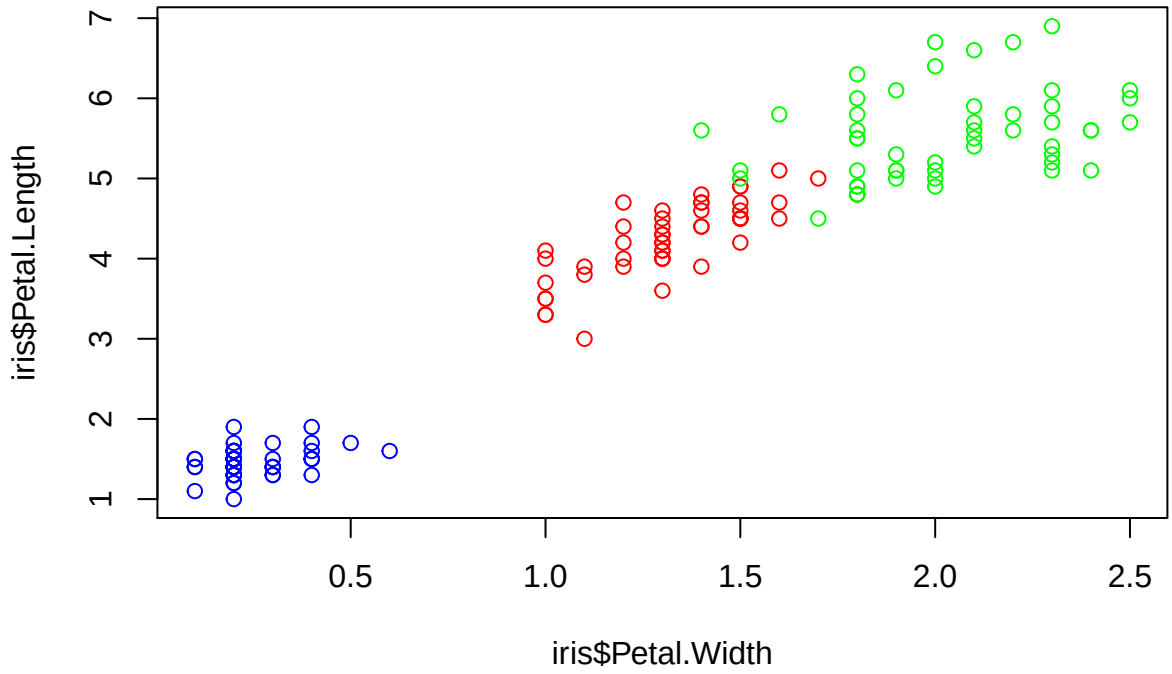

```
plot(iris$Sepal.Length, iris$Petal.Length, col=iris$Color)
```



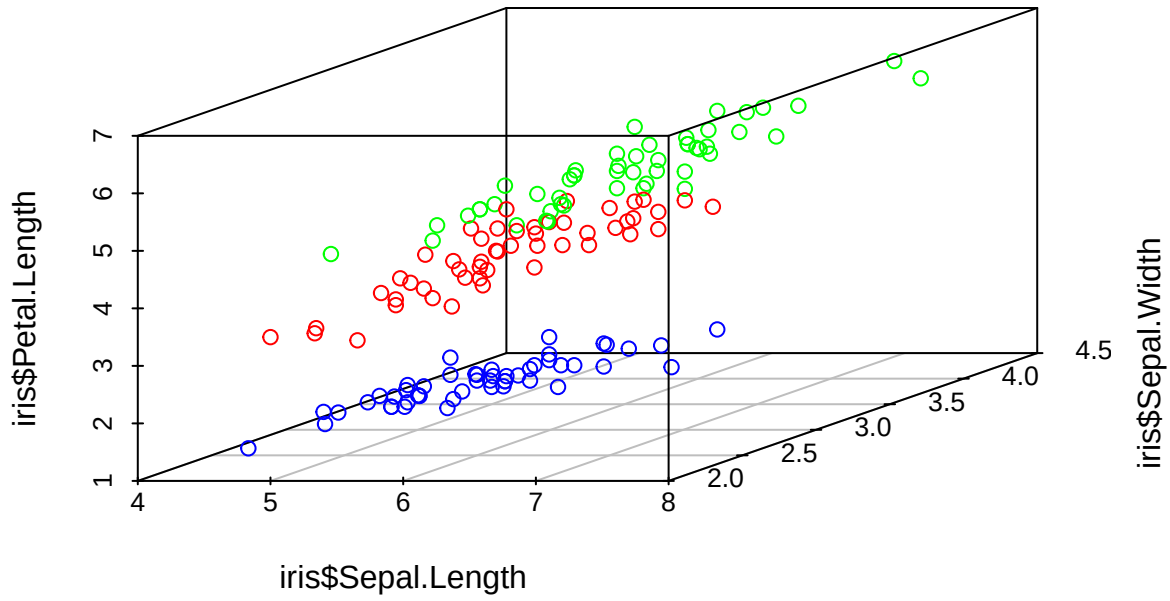

```
plot(iris$Sepal.Width, iris$Petal.Length, col=iris$Color)
```




```
plot(iris$Petal.Width, iris$Petal.Length, col=iris$Color)
```




```
library(scatterplot3d)
scatterplot3d(iris$Sepal.Length, iris$Sepal.Width, iris$Petal.Length, color=iris$Color)
```




```
scatterplot3d(iris$Sepal.Length, iris$Sepal.Width, iris$Petal.Length, color=iris$Color, angle = 20)
```