CM 2062 - Statistical Computing with R
Lab Sheet 5

Visualization of Data in R Using ggplot2 Package

ggplot2 is an open-source data visualization package for the statistical programming language R. ggplot2 allows you to create graphs for univariate and multivariate numerical and categorical data in a straightforward manner. It also allows for easy grouping and conditioning. It can generate complex plots create high quality graphics for publication.

Please go through the package manual "ggplot2.pdf" to get a better understanding of all the functions in the package.

Let's install the "ggplot2" package in R.

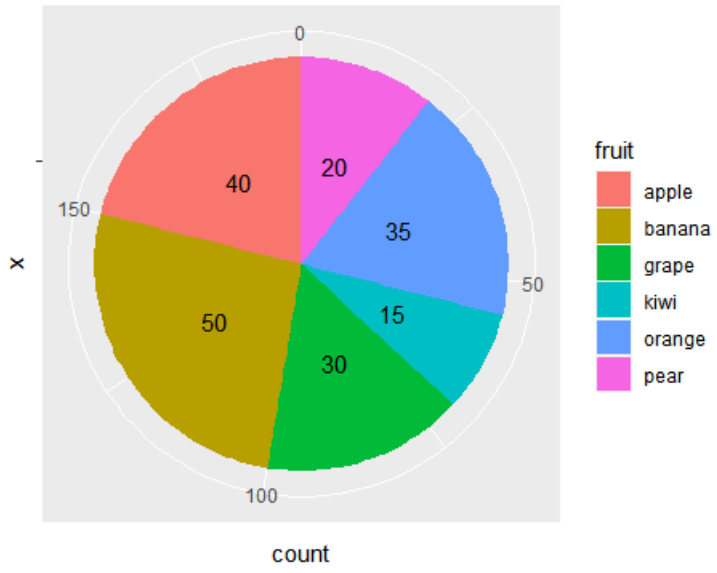
```
#installing and loading package
install.packages("ggplot2")
library(ggplot2)
```

Pie Chart in ggplot2 package

A pie chart in ggplot2 is a bar plot plus a polar coordinate.

Let's recall the favorite fruit example.

```
count <- c(40, 15, 30, 50, 20, 35 )
fruit <- c("apple","kiwi","grape", "banana", "pear", "orange")
# Let's create a dada frame call df for above data.
df <- data.frame(fruit , count)
df
#Let's creat a pie chart
pie <- ggplot(df, aes(x="", y=count, fill=fruit))+ geom_col() +
   coord_polar(theta ="y")
pie

#Adding text
pie <- pie+ geom_text(aes(label = count),
               position = position_stack(vjust = 0.5))
pie
```



It is possible to change manually the pie chart fill colors using the functions:
scale_fill_manual() : to use custom colors
scale_fill_brewer() : to use color palettes from RColorBrewer package
scale_fill_grey() : to use grey color palettes

```
# use brewer color palettes
pie1 <- pie + scale_fill_brewer()
pie1

pie2 <- pie  + scale_fill_brewer(palette="Dark2")
pie2

pie3 <- pie + scale_fill_brewer(palette="Blues")+theme_minimal()
pie3

# Use grey scale
pie4 <- pie + scale_fill_grey() + theme_minimal()
pie4

# Use custom color palettes
pie5 <- pie + scale_fill_manual(values=c( "#E69F00", "#56B4E9",
              "#BE2A3E", "#EC754A","#EACF65", "#3C8D53"))
pie5
```

Exercise Search more about changing legends, colours, themes ... etc in pie chart in ggplot2 package in R.

Bar Chart in ggplot2 package

The function **geom_bar()** can be used.

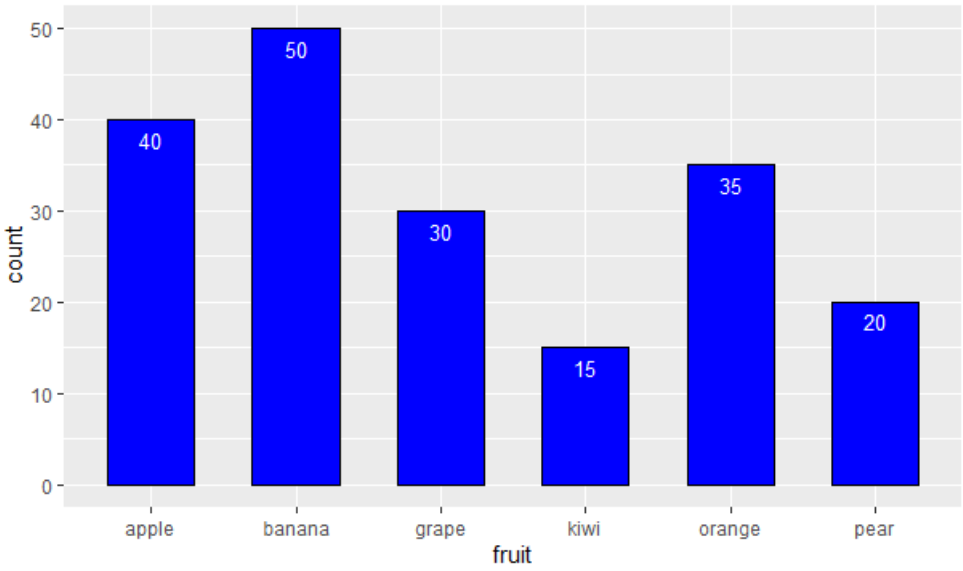
```
geom_bar(stat, fill, color, width)
```

Parameters
stat set the stat parameter to identify the mode.
fill represents color inside the bars.
color represents color of outlines of the bars.
width represents width of the bars.

Let's recall the favorite fruit example.

```
library(ggplot2)
count <- c(40, 15, 30, 50, 20, 35 )
fruit <- c("apple","kiwi","grape", "banana", "pear", "orange")
df <- data.frame(fruit, count)

# Basic vertical barplot
Bar <- ggplot(data=df, aes(x=fruit, y=count))+
  geom_bar(stat="identity")
Bar

# Horizontal bar plot
Bar <- Bar+coord_flip()
Bar

#changing the color of the Bar plot

Bar <- ggplot(data=df, aes(x=fruit, y=count))+
  geom_bar(stat="identity", fill="blue")
Bar

# adding outline color
Bar <- ggplot(data=df, aes(x=fruit, y=count))+
  geom_bar(stat="identity", color= "black", fill="blue")
Bar

# Change the width of bars
Bar <- ggplot(data=df, aes(x=fruit, y=count))+
  geom_bar(stat="identity", color= "black", fill="blue", width=0.6)
Bar

#Bar plot with labels
```

```
Bar <- ggplot(data=df, aes(x=fruit, y=count))+
  geom_bar(stat="identity", color="black", fill="blue", width=0.6)+
  geom_text(aes(label=count), vjust=1.6, color="white", size=3.5)
Bar
```

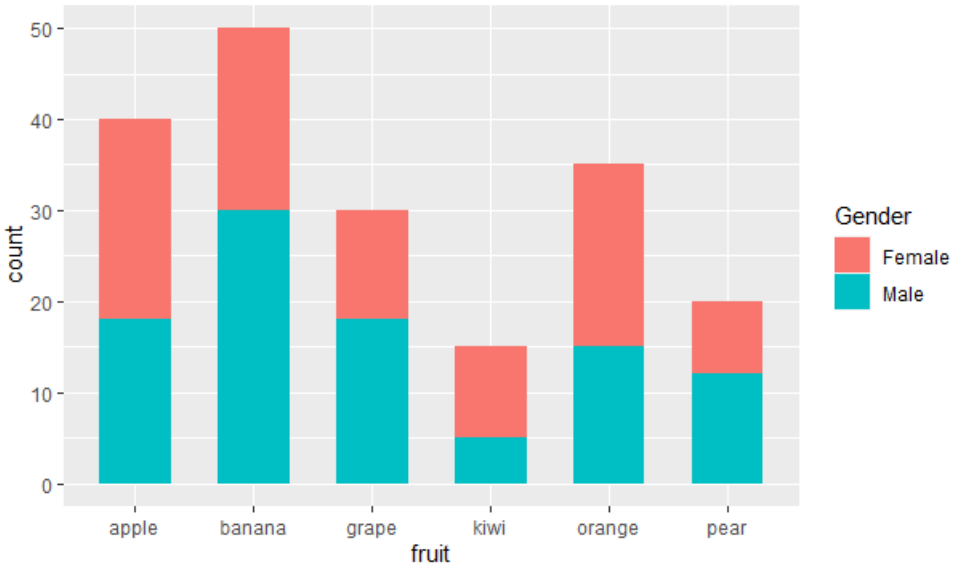


Stacked Bar Chart in ggplot2 package

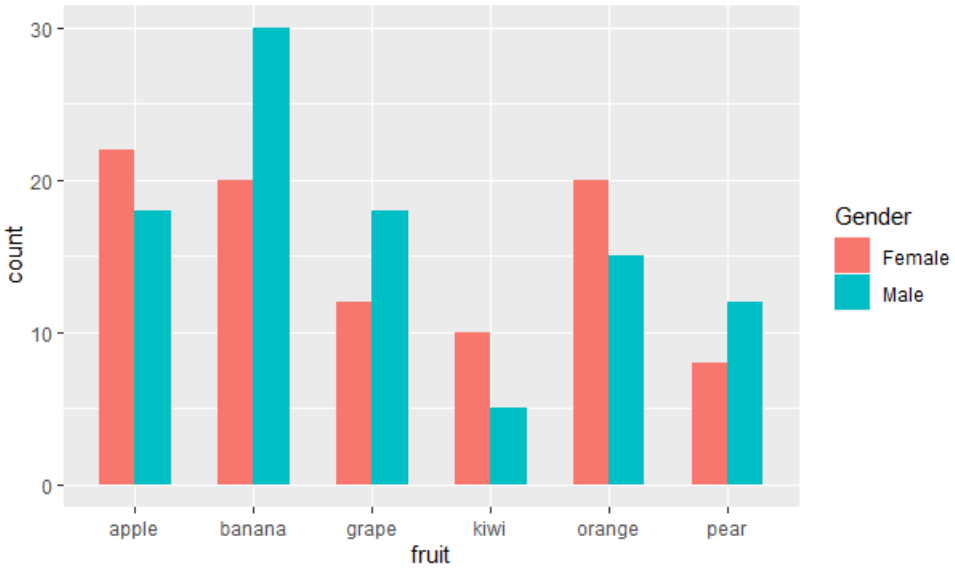
Let's recall the favorite fruit example by gender.

Let's create the data frame call "df1" first.

```
count <- c(18,5,18,30,12,15,22,10,12,20,8,20 )
fruit <- rep(c("apple","kiwi","grape", "banana", "pear", "orange"),2)
Gender <- rep(c("Male", "Female"), each=6)
df1 <- data.frame(Gender, fruit, count)
df1

#Stacked Bar Chart
Bar1 <- ggplot(data=df1, aes(x=fruit, y=count, fill=Gender)) +
  geom_bar(stat="identity", width=0.6)
Bar1
```



Group Bar Chart in ggplot2 package



Exercise

Consider above Stacked bar chart and Grouped bar chart. change their colours. Add a suitable topic to the chart. Add value labels.

Visualization of Numerical Data in R using ggplot2 package

Line chart in ggplot2 package

The functions **geom_line()** can be used.

x value (for x axis) can be :
date : for a time series data
texts
discrete numeric values
continuous numeric values

Recall the Temperature example

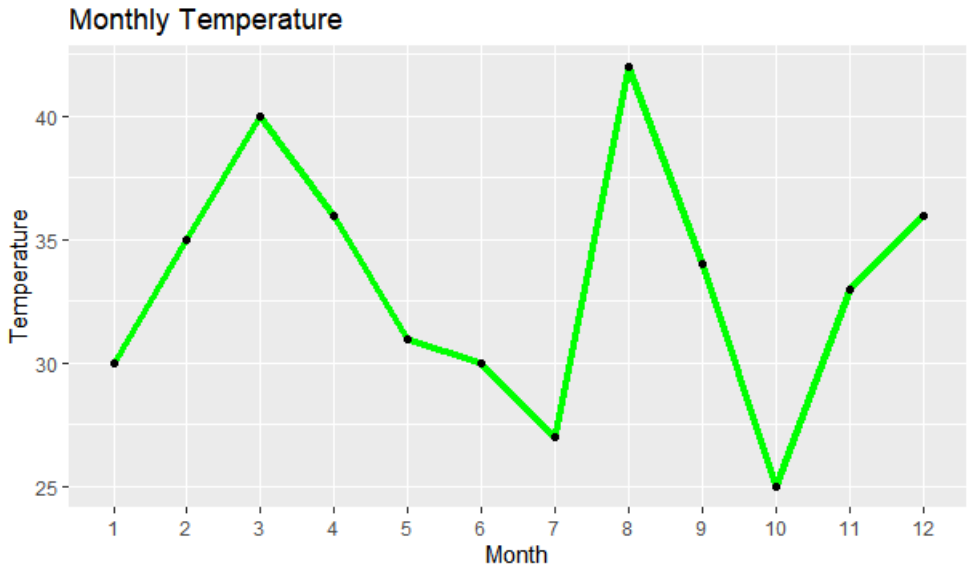
```
Temp <- c(30, 35, 40, 36, 31, 30 ,27, 42, 34, 25, 33, 36 )
Month <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10,  11, 12)
Month <- as.factor(Month)
df1 <- data.frame (Month, Temp)

# Basic Line
library(ggplot2)
ggplot(data=df1, aes(x=Month, y=Temp, group=1)) +
  geom_line()+
  geom_point()
```

Formating Line
For this, the command linetype is used. ggplot2 provides various line types. For example : dotted, two dash, dashed, etc. This attribute is passed with a required value.

```
# Format the line type
ggplot(data=df1, aes(x=Month, y=Temp, group=1)) +
  geom_line(linetype = "dotted")+
  geom_point()

# Format the line color
ggplot(data=df1, aes(x=Month, y=Temp, group=1)) +
  geom_line( color="green")+
  geom_point()

# Format the line size
ggplot(data=df1, aes(x=Month, y=Temp, group=1)) +
  geom_line( color="green",size=1.5)+
  geom_point()
```

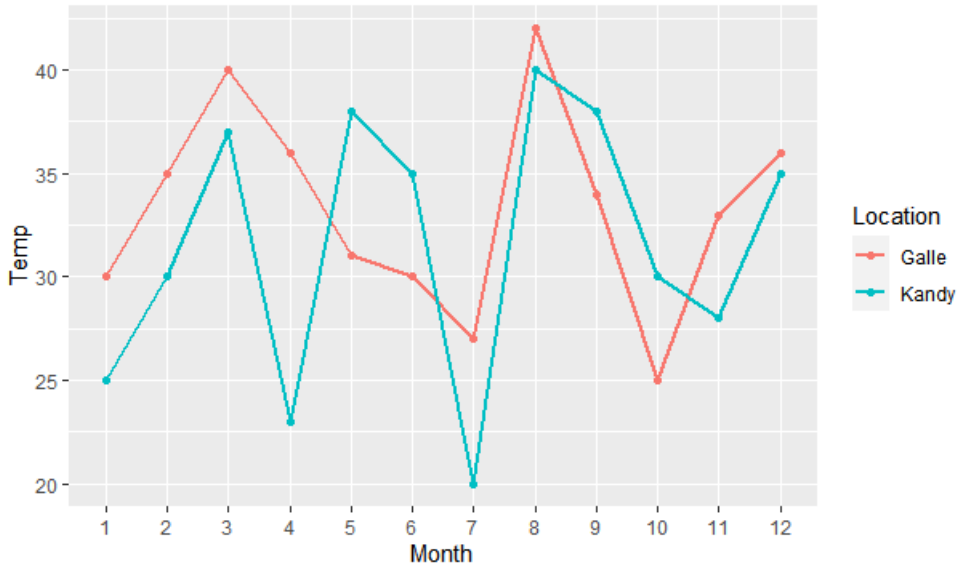
Adding Chart Title, Axis Title
ggtitle() with the appropriate title can be used to add chart title and **labs** again with appropriate input can be used to add axes title.

```
# Adding titles
L <- ggplot(data=df1, aes(x=Month, y=Temp, group=1)) +
  geom_line( color="green",size=1.5)+
  geom_point()

L + ggtitle("Monthly Temperature")+
  labs(x="Month",y="Temperature")
```



Plotting Multiple lines

```
Month <- rep(c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10,  11, 12), 2)
Month <- as.factor(Month)
Location <- rep(c("Galle", "Kandy"), each=12)
Temp <- c(30, 35, 40, 36, 31, 30 ,27, 42, 34, 25, 33, 36, 25,
                30, 37, 23, 38, 35, 20,  40, 38, 30, 28, 35 )
df2 <- data.frame (Location, Month, Temp)

# Plotting line with multiple groups
ggplot(data=df2, aes(x=Month, y=Temp, group=Location)) +
  geom_line(aes(color=Location), size=1)+
  geom_point(aes(color=Location))
```



Exercise

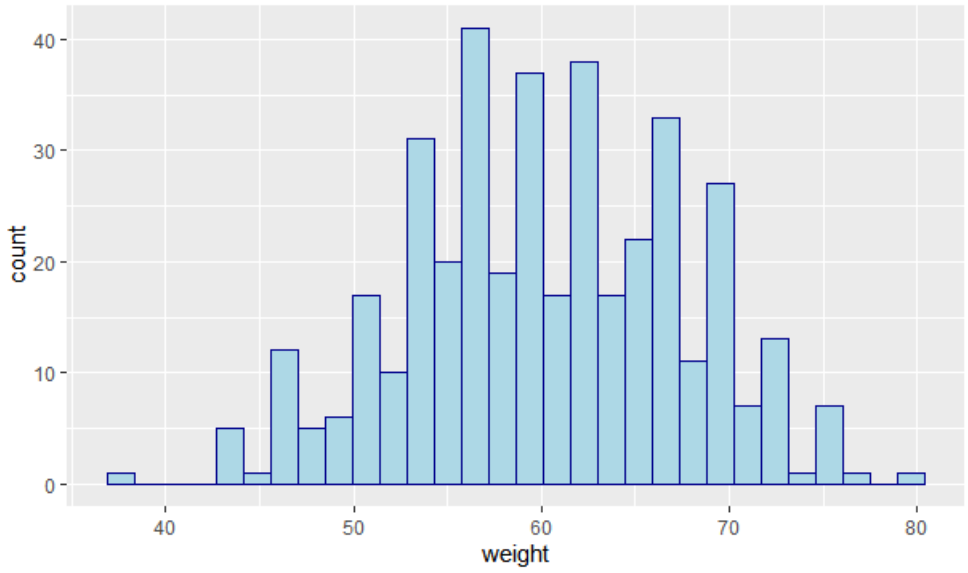
Change the line colors, legend position, add topic, labels ... etc to above multiple line chart.

Histogram in ggplot2 package

The function **geom_histogram()** is used.

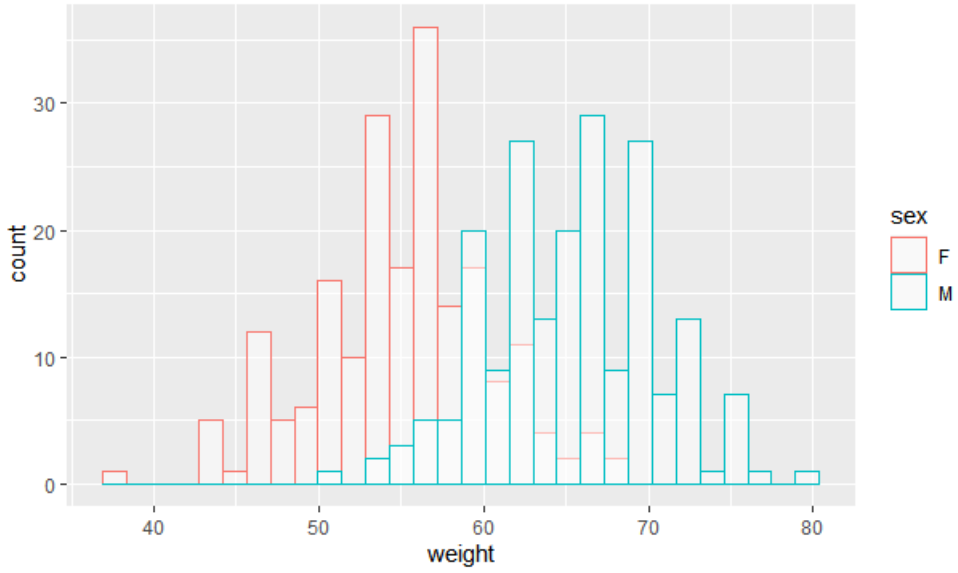
Example

```
df <- data.frame(
  sex=factor(rep(c("F", "M"), each=200)),
  weight=round(c(rnorm(200, mean=55, sd=5), rnorm(200, mean=65, sd=5)))
)
library(ggplot2)
# Basic histogram
ggplot(df, aes(x=weight)) + geom_histogram()
# Change the width of bins
ggplot(df, aes(x=weight)) +
  geom_histogram(binwidth=2)
# Change colors
p <-ggplot(df, aes(x=weight)) +
  geom_histogram(color="black", fill="white")
p

# Change line color and fill color
ggplot(df, aes(x=weight))+
  geom_histogram(color="darkblue", fill="lightblue")
```



Also, we can create the group wise histograms in the same plot.

```
# Change histogram plot line colors by groups
ggplot(df, aes(x=weight, color=sex)) +
  geom_histogram(fill="white")

# Overlaid histograms
P <- ggplot(df, aes(x=weight, color=sex)) +
  geom_histogram(fill="white", alpha=0.5, position="identity")
P
```



Exercise

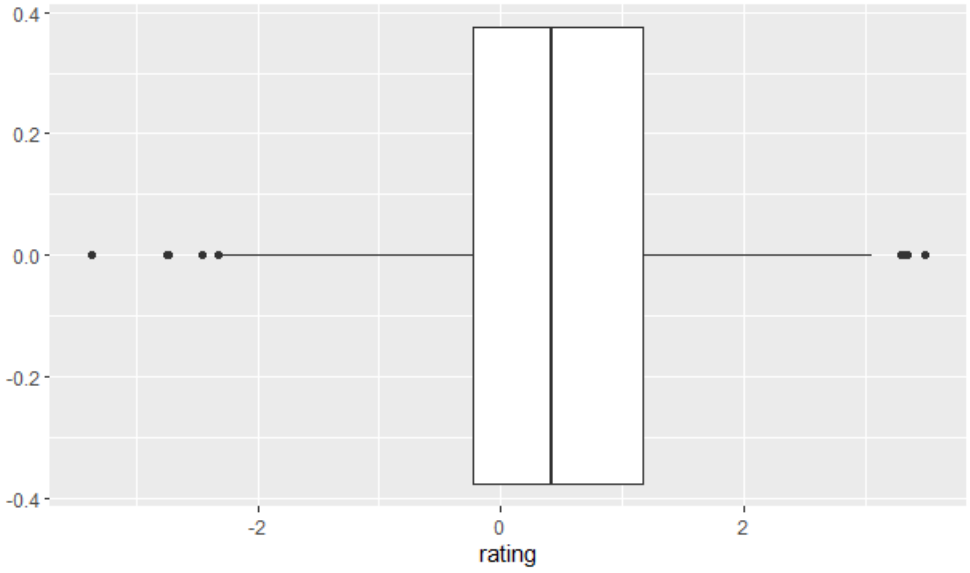
Change fill colours, line colours, legend positions .... etc in the above histograms.

Box Plot in ggplot2 package

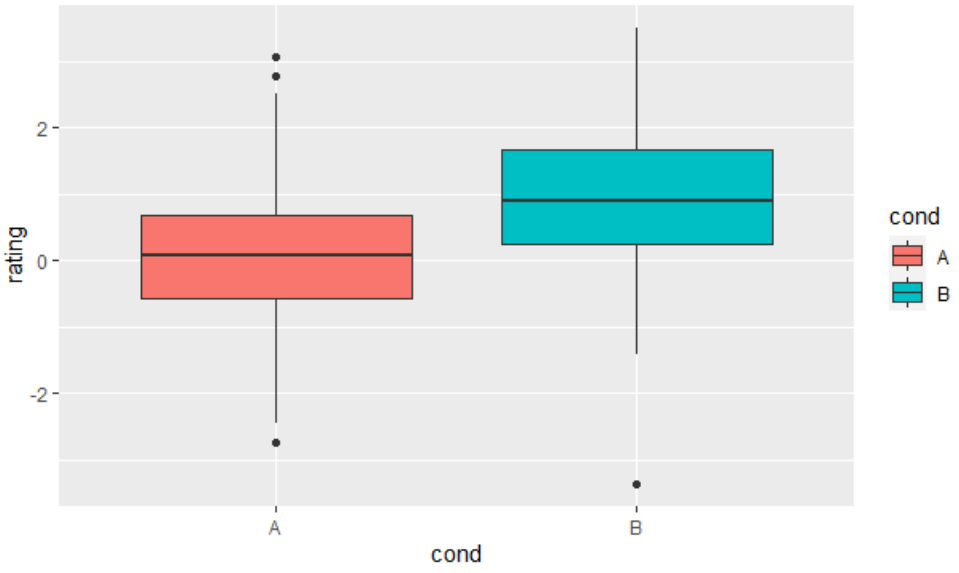
The function geom_boxplot() is used.

Example

```
data <- data.frame(cond = factor(rep(c("A","B"), each=200)),
        rating = c(rnorm(200),rnorm(200, mean=.8)))
library(ggplot2)
ggplot(data, aes(y=rating)) + geom_boxplot()
#Horizontal Box Plot
ggplot(data, aes(x=rating)) + geom_boxplot()
```




```
#Box plot with groups
ggplot(data, aes(x=cond, y=rating)) + geom_boxplot()
#Horizontal Box Plots
ggplot(data, aes(x=cond, y=rating)) + geom_boxplot() + coord_flip()
#Colored Boxplot
ggplot(data, aes(x=cond, y=rating, color=cond)) + geom_boxplot()
ggplot(data, aes(x=cond, y=rating, fill=cond)) + geom_boxplot()
```



Exercise 1

Change the fill colours, legend position and add a title ... etc to the above box plots.

Exercise 2

Use the "mtcars" data set in R.
(i) Draw a box plot for "mpg" variable and give a suitable title to the box plot and fill the box plot with a suitable colour.
(ii) Draw a box plot with groups for "mpg" variable by taking "cyl" as the grouping variable. Give a suitable title and fill the box plots with colours in gray scale.

Scatter Plot in ggplot2 package

The function geom_point() is used.

Let's consider the "mtcars" data set.

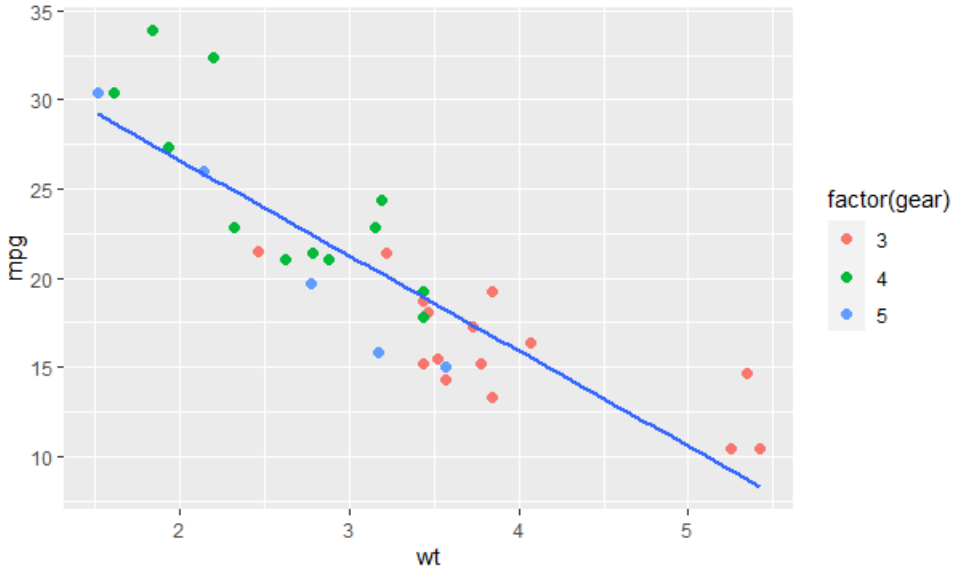
```
library(ggplot2)
ggplot(mtcars, aes(x = wt, y = mpg)) + geom_point()
# Change color, shape and size
ggplot(mtcars, aes(x = wt, y = mpg)) +
    geom_point(color = "blue", size = 2, shape = 6)
```

Add regression line

geom_smooth() function is used.

```
ggplot(mtcars, aes(x = wt, y = mpg)) + geom_point() +
     geom_smooth(method = lm, se = FALSE)
#Scatter plot with groups
```

```
ggplot(mtcars, aes(x = wt, y = mpg)) +
    geom_point(aes(color = factor(gear)), size=2) +
    geom_smooth(method = lm, se = FALSE)
```



Exercise

Add figure title, change the colours, shapes ... etc in the above scatter plot.

Scatter plot Matrices in ggplot2 package

Exercise

Use "mtcars" data set. Create a scatter plot matrices for wt, mpg, disp, cyl variables using ggplot2.