Activity 7 - Aryan Khandelwal

**Question 1:** We shall generate several figures by using generic plot function.

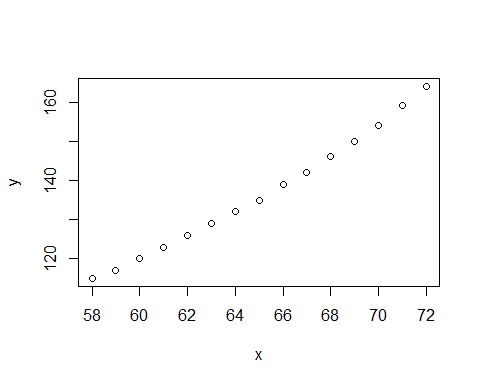
1. Let’s load women data which is saved in dataset R package. And check what variables are included in this dataset by using names Weightfunction.

data(women, package="datasets")  
names(women)

## [1] "height" "weight"

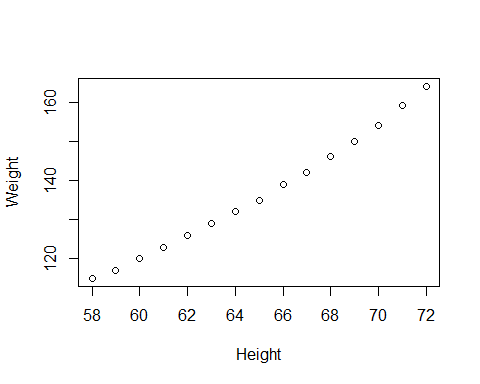
1. Let’s generate a basic plot by displaying x=height, y=weight.

x <- women$height  
y <- women$weight  
plot(x, y)



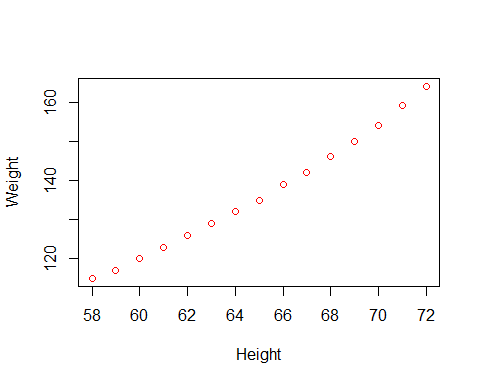
1. Let’s add labels for x and y, e.g., x-label = Height, y-label = Weight.

plot(x, y, xlab = "Height", ylab = "Weight")



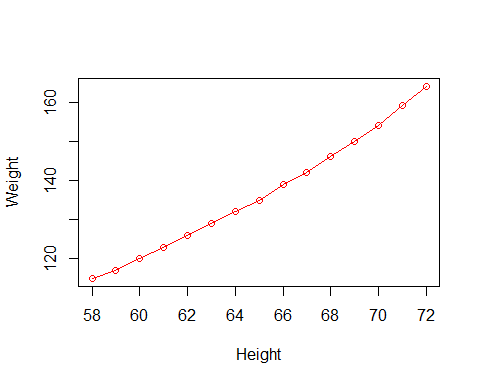
1. Let’s change the color of the points to be red.

plot(x, y, xlab = "Height", ylab = "Weight", col = "red")



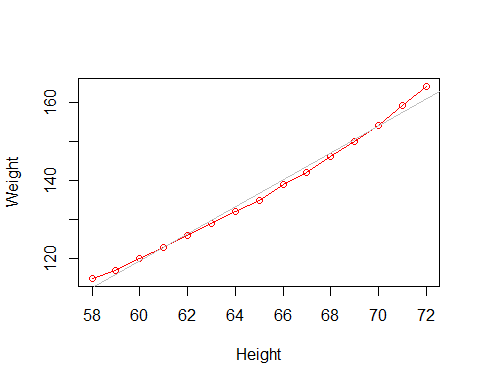
1. Let’s generate a different type of a plot which displays both the points and lines. Hint: turn on type argument in a generic plot() function.

plot(x, y, xlab = "Height", ylab = "Weight", col = "red", type = "o")



1. Let’s add a regression line to the previous plot and let this regression line has gray color.

plot(x, y, xlab = "Height", ylab = "Weight", col = "red", type = "o")  
abline(lm(y~x, data = women), col= "gray" )



**Question 2:** We will use ggplot2 R package to generate similar plots. Load ggplot2 R package.

install.packages("ggplot2", repos = "http://cran.us.r-project.org")

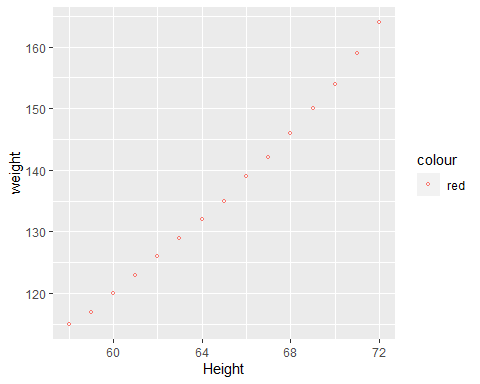
## Installing package into 'C:/Users/khand/OneDrive'  
## (as 'lib' is unspecified)

## package 'ggplot2' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\khand\AppData\Local\Temp\Rtmpwllsck\downloaded\_packages

library(ggplot2)

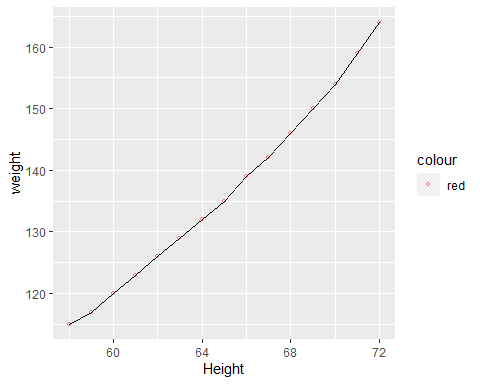
1. Use ggplot() function to generate a (point) plot with x=Height, y=Weight. Add labels for x and y like the graph in Question 1 (c). And let the points be red. Also, let pch=1 and size=1 for the points. Hint: Use ggplot(), geom\_point(), ylab, and xlab (See Visualization slides)

ggplot(women, aes(x = height, y = weight, color = "red")) +  
 geom\_point(pch=1, size = 1) +  
 xlab("Height") + ylab("weight")



1. Let’s add lines to the previous graph and connect the points. Let the lines be black color. Hint: Use geom\_line() (See Visualization slides)

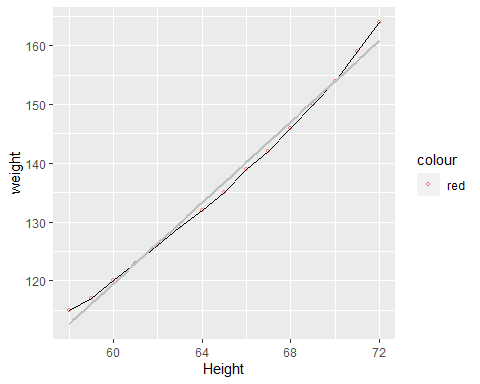
ggplot(women, aes(x = height, y = weight, color = "red")) +  
 geom\_point(pch=1, size = 1) +  
 xlab("Height") + ylab("weight") +  
 geom\_line(color = "black")



1. Let’s add a regression line to the previous graph and let this regression line has gray color. Hint: Use geom\_smooth() (See Visualization slides)

ggplot(women, aes(x = height, y = weight, color = "red")) +  
 geom\_point(pch=1, size = 1) +  
 xlab("Height") + ylab("weight") +  
 geom\_line(color = "black")+  
geom\_smooth(method = "lm", color = "gray", se = FALSE)

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



1. Let’s add a smooth line using loess to the previous graph and let this regression line has blue color. Hint: Use geom\_smooth() (See Visualization slides)

ggplot(women, aes(x = height, y = weight, color = "red")) +  
 geom\_point(pch=1, size = 1) +  
 xlab("Height") + ylab("weight") +  
 geom\_line(color = "black") +  
geom\_smooth(method = "lm", color = "gray", se = FALSE) +  
 geom\_smooth(method = "loess", color = "blue", se = FALSE)

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

