Lance Dacy

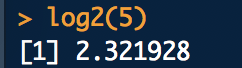
MSDS 6306.Assignment1

May 16, 2018

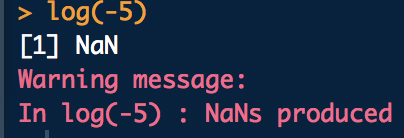
1. Basic Math
   1. Log of a positive number: log(5)



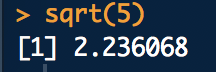
* 1. What is the default base for the log function? Calculate the log of your previous number with a different base? 10. Calculate the log of your previous number with a different base: log2(5)



* 1. The log of a negative number. (explain the answer): log(-5)
     1. The log of a negative number produced NaN (undefined, since the base has to be > 0 to be a real number)



* 1. The square-root of a positive number: sqrt(5)



1. Random Number Generation
   1. Create a vector of 15 standard normal random variables. Calculate its mean and

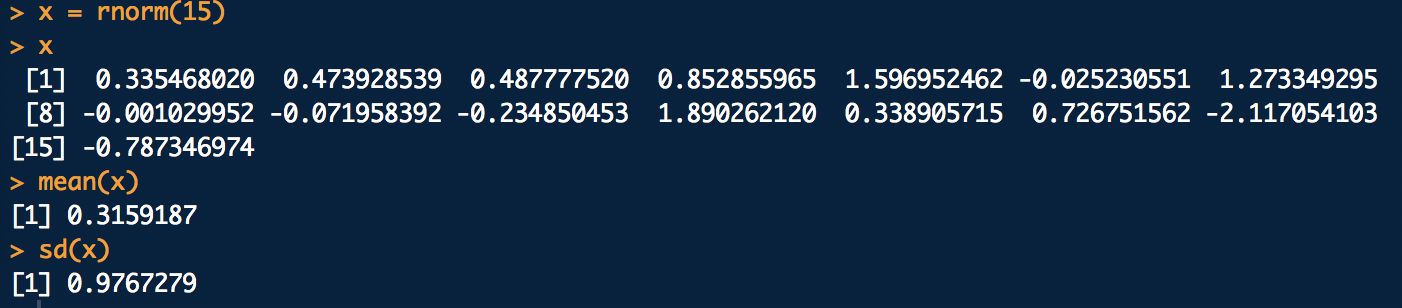
SD (Standard Deviation).

x = rnorm(15)

x

mean(x)

sd(x)



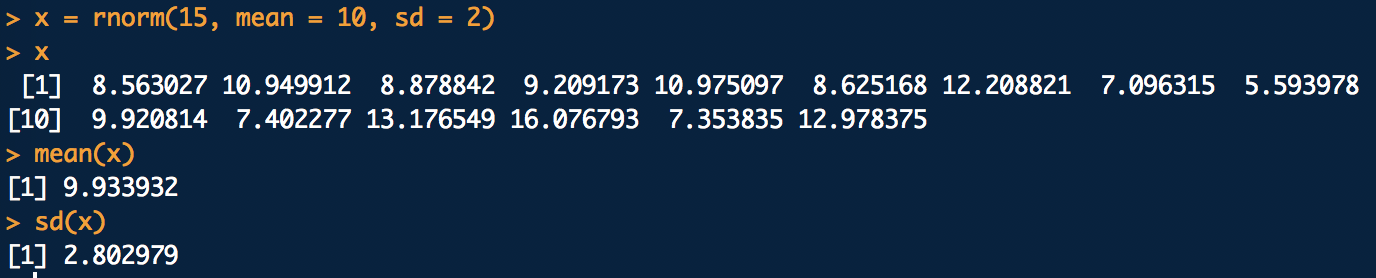
* 1. Change the mean to 10 and the SD to 2 and recalculate the vector of 15 random normal variables. Calculate its mean and SD.

x = rnorm(15, mean = 10, sd = 2)

x

mean(x)

sd(x)

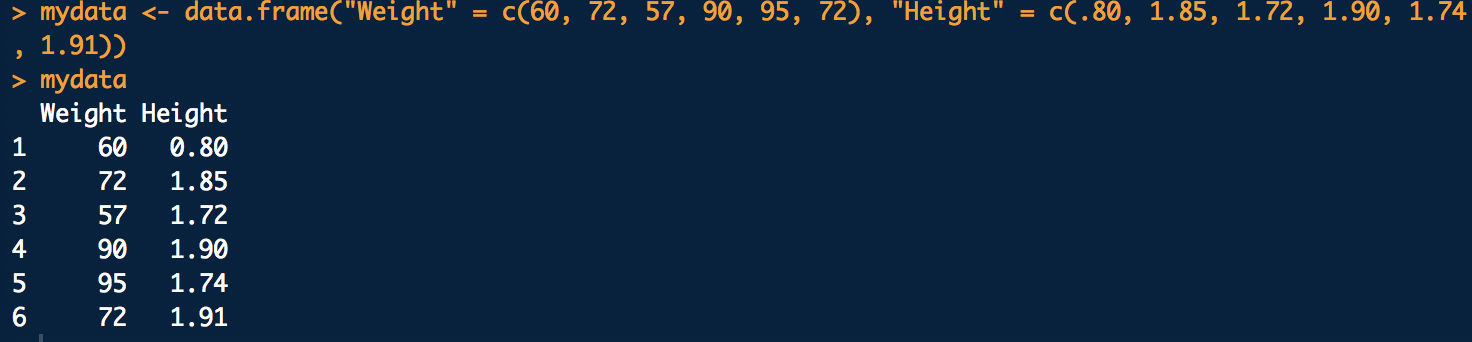


* 1. Why are the means and SD not exactly the same as the means and SDs specified in the function? Because the numbers are a randomly generated set.

1. Vector Operations
   1. The weights of 6 individuals in kg are 60, 72, 57, 90, 95, 72.
   2. Their heights (in m) are 1.80, 1.85, 1.72, 1.90, 1.74, 1.91.
   3. Enter these vectors into R.

mydata <- data.frame("Weight" = c(60, 72, 57, 90, 95, 72), "Height" = c(.80, 1.85, 1.72, 1.90, 1.74, 1.91))

mydata



* 1. Create a scatterplot of weight vs. height. Interpret the scatterplot. It appears that height is somewhat proportional to weight.

#Assign the data from the data above to the scatter plot

input <- mydata[,c("Weight","Height")]

print(head(input))

#File name

png(file = "scatterplot.png")

# Plot the chart

plot(x = input$"Weight",y = input$"Height",

xlab = "Weight",

ylab = "Height",

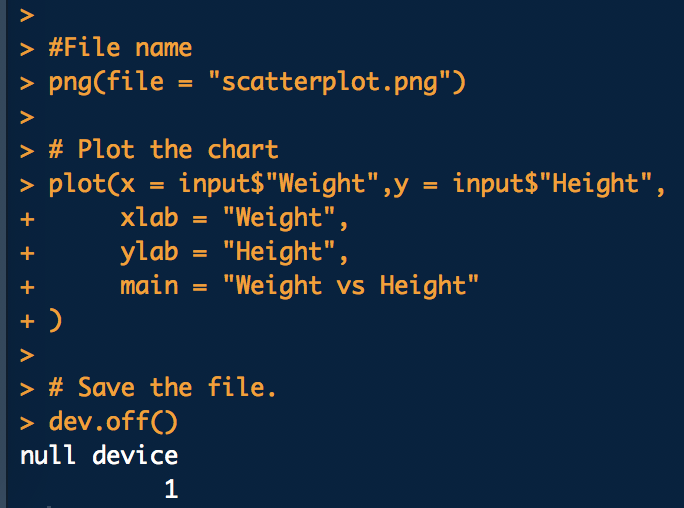
main = "Weight vs Height"

)

# Save the file.

dev.off()

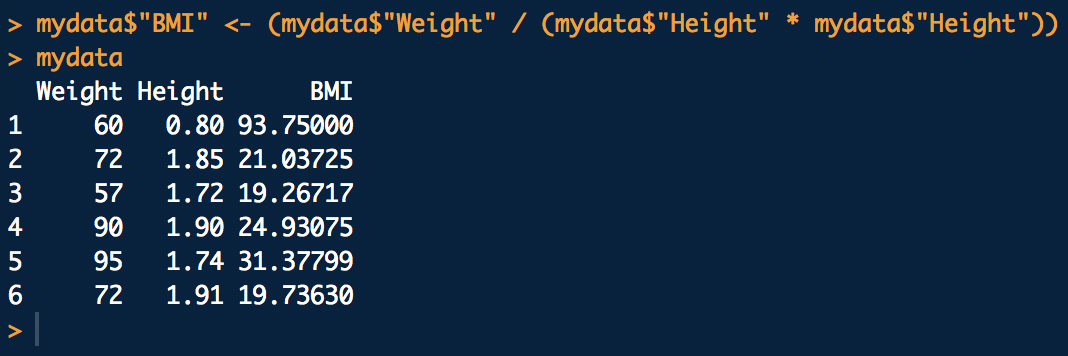




* 1. Calculate the BMI for each individual. (BMI = weight in kg divided by the square of the height in m)

mydata$"BMI" <- (mydata$"Weight" / (mydata$"Height" \* mydata$"Height"))

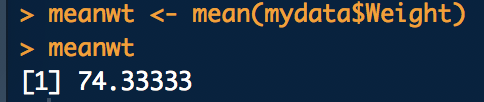
mydata



* 1. Calculate the mean for weight.

meanwt <- mean(mydata$Weight)

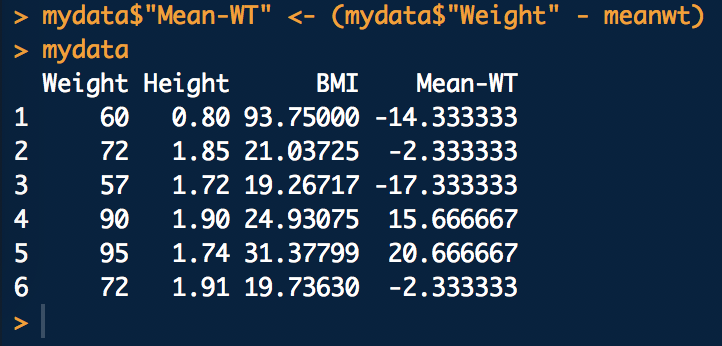
meanwt



* 1. Subtract the mean from each value of weight.

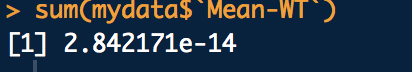
mydata$"Mean-WT" <- (mydata$"Weight" - meanwt)

mydata



* 1. Sum the result. Now you know why we square the deviations from the mean to calculate a standard deviation!

sum(mydata$`Mean-WT`)



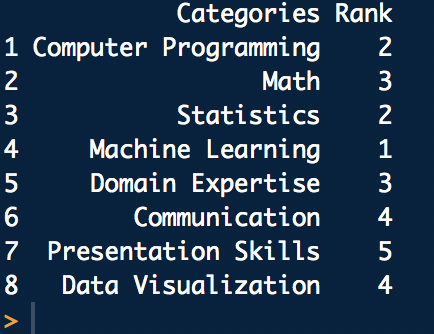
1. Enter your data science profile into R as a data frame with two columns. Call it by your first name. The categories are computer programming, math, statistics, machine learning, domain expertise, communication and presentation skills, and data visualization. Your ranking for each category 1 5, with 5 as best. Create a bar graph of your data science profile. When you submit your work, please submit your code, including the data entry piece.

Lance <- data.frame("Categories" = c("Computer Programming", "Math", "Statistics",

"Machine Learning", "Domain Expertise",

"Communication", "Presentation Skills", "Data Visualization"), "Rank" = c(2,3,2,1,3,4,5,4))

Lance



#4. The bar chart

#File name

png(file ="barchart.png")

barplot(Lance$Rank,names.arg=Lance$Categories,ylab="Ranking",col="blue",

main="Lance's Profile",border="red", las=2, cex.axis = .50, cex.names = .50)

# Save the file.

dev.off()

