DS 6306 – HW 1

Brandon Croom

## Question 1

1. The log of a positive number

log(20)

## [1] 2.995732

1. What is the default base for the log function? Calculate the log of your previous number with a different base The default base for the log function is base 10. Calculating the log for 20 in base 2 is as follows:

log(20,base=2)

## [1] 4.321928

1. The log of a negative number (explain the answer)

log(-20)

## Warning in log(-20): NaNs produced

## [1] NaN

The result produces a NaN value. This indicates that the results from taking a negative log are invalid in R

1. The square-root of a positive number

sqrt(20)

## [1] 4.472136

## Question 2 - Random number generation

1. Create a vector of 15 standard normal random variables. Calculate its mean and SD (standard deviation)

#Define the random\_normal vector with 15 values with default mean and sd  
random\_normal = rnorm(15)  
  
#show output of random\_normal vector  
random\_normal

## [1] 0.64127813 0.04990951 0.52089374 0.30227564 1.17597252  
## [6] -2.22918715 1.00168032 1.67767886 -0.79758408 -0.04902578  
## [11] -0.46553815 -0.02959952 -0.66863542 0.39320752 0.29778894

#Calculate the mean of random\_normal vector  
mean(random\_normal)

## [1] 0.1214077

#Calculate the standard deviation of random\_normal vector  
sd(random\_normal)

## [1] 0.9362539

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

#Define the random\_normal vector with 15 values with mean = 10 and sd = 2  
random\_normal = rnorm(15, mean=10, sd=2)  
  
#show output of random\_normal vector  
random\_normal

## [1] 11.714601 6.758246 6.881885 8.935030 9.586407 9.589948 11.089989  
## [8] 9.643295 7.702153 10.123205 12.720956 7.884892 10.691249 9.663525  
## [15] 8.418430

#Calculate the mean of random\_normal vector  
mean(random\_normal)

## [1] 9.426921

#Calculate the standard deviation of random\_normal vector  
sd(random\_normal)

## [1] 1.717532

1. Why are the means and SD not exactly the same as the means and SDs specified in the function?

The means and SD are not exactly the same as the means specified due to requesting random variables in the vector. The random number generation may not be able to get the exact value but will get close

## Question 3 - Vector Operations

1. The weights of 6 individuals in k6 are 60, 72, 57, 90, 95, 72
2. The heights (in m) are 1.80, 1.85, 1.72, 1.90, 1.74, 1.91
3. Enter these vectors into R

#Create vectors called weight and height respectively to house the data in a and b above  
weight = c(60, 72, 57, 90, 95, 72)  
  
height = c(1.80, 1.85, 1.72, 1.90, 1.74, 1.91)  
  
#Output weight and height vectors to verify output  
weight

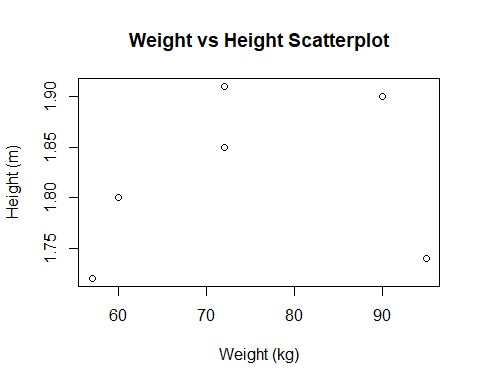
## [1] 60 72 57 90 95 72

height

## [1] 1.80 1.85 1.72 1.90 1.74 1.91

1. Create a scatterplot of weight vs. height. Interpret the scatterplot

plot(weight,height, main="Weight vs Height Scatterplot", xlab="Weight (kg)", ylab="Height (m)")



The scatterplot shows in general, that higher weights translate to taller individuals

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

#define vector BMI to hold BMI values  
BMI = weight / (height ^ 2)  
  
#Output BMI calculation  
BMI

## [1] 18.51852 21.03725 19.26717 24.93075 31.37799 19.73630

1. Calculate the mean for weight

#create a variable to hold the mean of the weight  
weight\_mean = mean(weight)  
  
#Display the mean of the weight vector  
weight\_mean

## [1] 74.33333

1. Subtract the mean from each value of weight

#define a variable to hold calculation  
weight\_sub\_mean = weight - weight\_mean  
  
#display the weight\_sub\_mean variable  
weight\_sub\_mean

## [1] -14.333333 -2.333333 -17.333333 15.666667 20.666667 -2.333333

1. Sum the result.

sum(weight\_sub\_mean)

## [1] 2.842171e-14

## Question 4 - Data Science Profile

#Build out data frame called Brandon and store personal ranking for the data science profile  
#Abbreviations for data science skills:  
# Computer Programming = CompProg  
# Math = Math  
# Statistics = Stats  
# Machine Learning = ML  
# Domain Expetise = DomainExp  
# Communications and Presentation Skills = CommPres  
# Data Visualization = DataViz  
  
Brandon = data.frame(  
 ds\_category = c("CompProg", "Math", "Stats", "ML", "DomainExp", "CommPres", "DataViz"),  
 ds\_rank = c(3,3,3,2,4,4,2)  
)  
  
#Display the results of Brandon in a bar graph  
 barplot(Brandon$ds\_rank,main="Data Science Profile - Brandon Croom",names.arg=Brandon$ds\_category, cex.names=0.5, space = 1)

