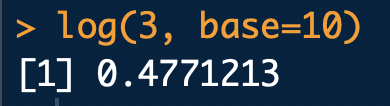
1. Basic Math
   1. log(3)

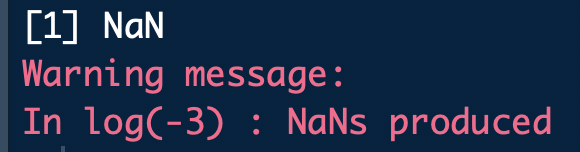


* 1. The default base of the log function is exp(1) or e.

log(3, base=10)



* 1. log(-3)



This produces a NaN because you cannot take the logarithm of a negative number. A NaN means it is "not a number".

* 1. sqrt(4)

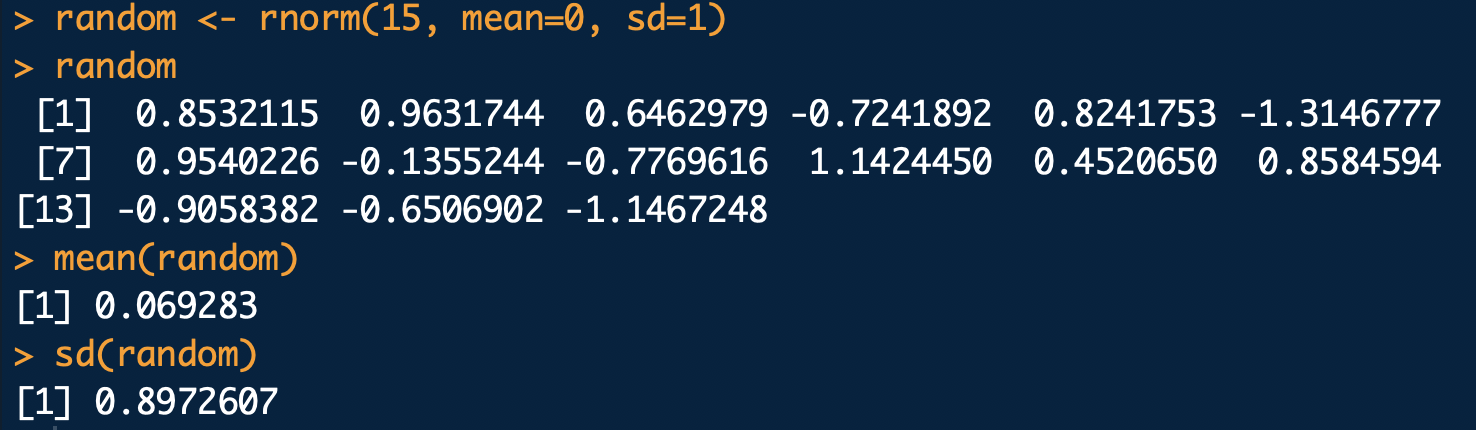


1. Random Number Generation
   1. random <- rnorm(15, mean=0, sd=1)

random

mean(random)

sd(random)

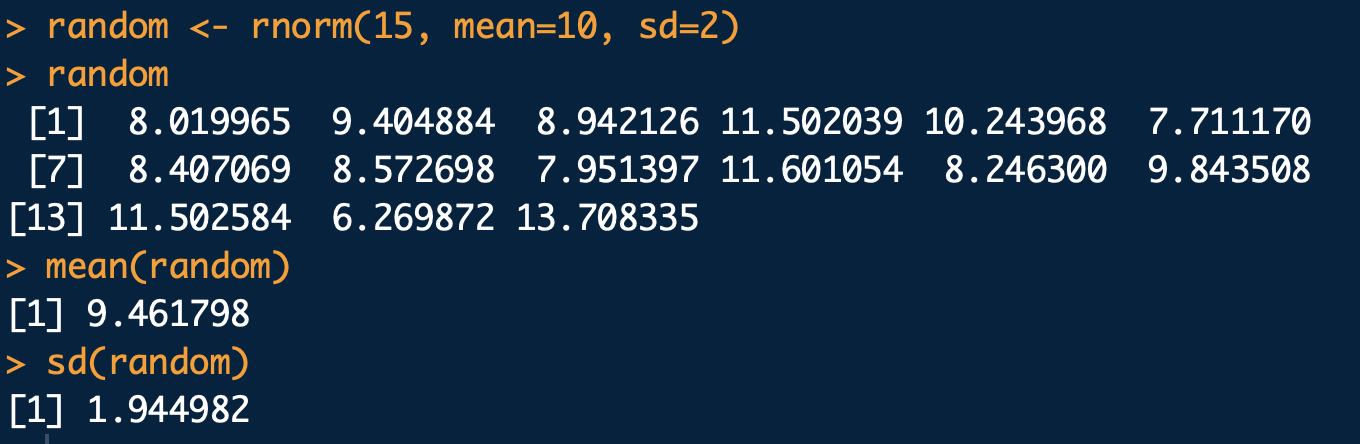


* 1. random <- rnorm(15, mean=10, sd=2)

random

mean(random)

sd(random)



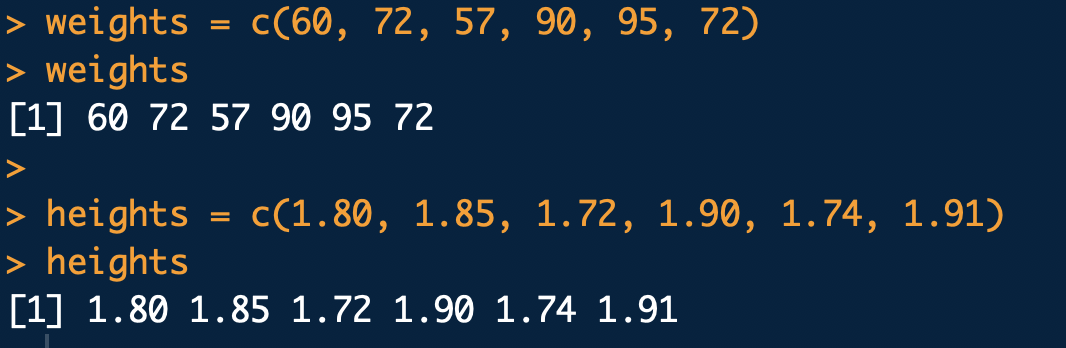
* 1. The means and standard deviations for both of the above problems for the samples are close to the means and standard deviations of the distributions they came from. However, they are not exactly the same because samples will never have exactly the same distribution and thus will never have the exact same mean and standard deviation.

1. Vector Operations
   1. …
   2. …
   3. weights = c(60, 72, 57, 90, 95, 72)

weights

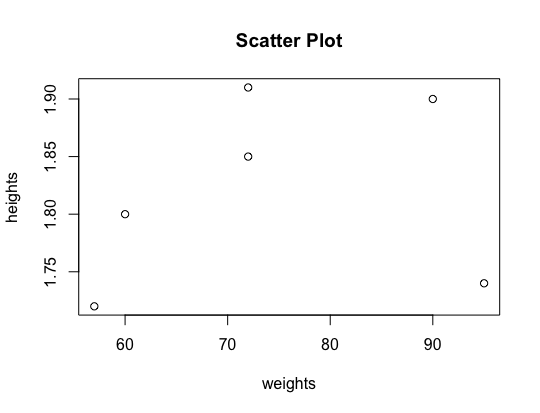
heights = c(1.80, 1.85, 1.72, 1.90, 1.74, 1.91)

heights



* 1. plot(x = weights, y=heights, xlab = 'weights', ylab='heights',

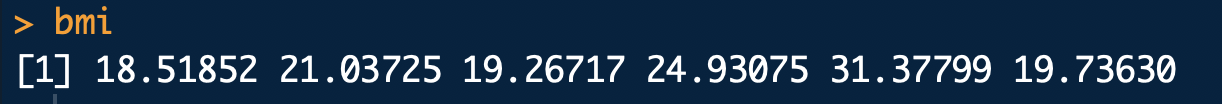
main='Scatter Plot')



Based on this data, there seems to be a positive relationship between heights and weights, but there is an outlier in the bottom right corner.

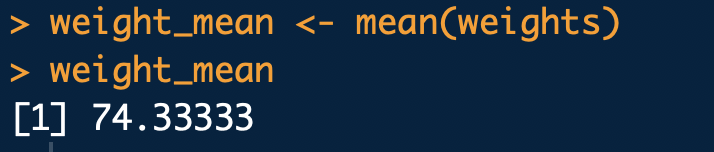
* 1. bmi = weights / (heights^2)

bmi



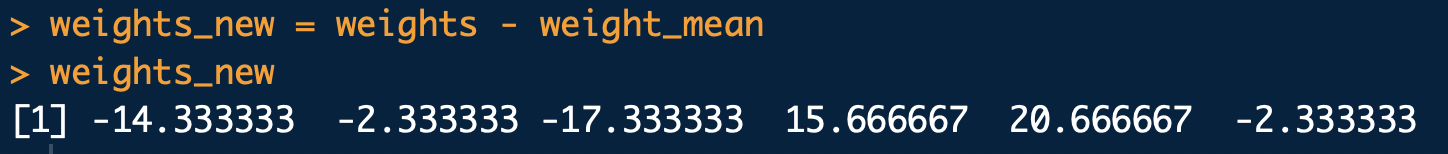
* 1. weight\_mean <- mean(weights)

weight\_mean



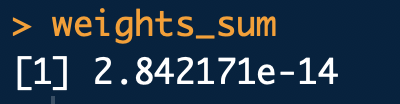
* 1. weights\_new = weights - weight\_mean

weights\_new



* 1. weights\_sum = sum(weights\_new)

weights\_sum



1. categories = c('computer programming', 'math',

'statistics', 'machine learning',

'domain expertise', 'communication/presentation',

'data visualization')

rankings = c(3.5, 2.5, 2.5, 2, 4, 3.5, 3.5)

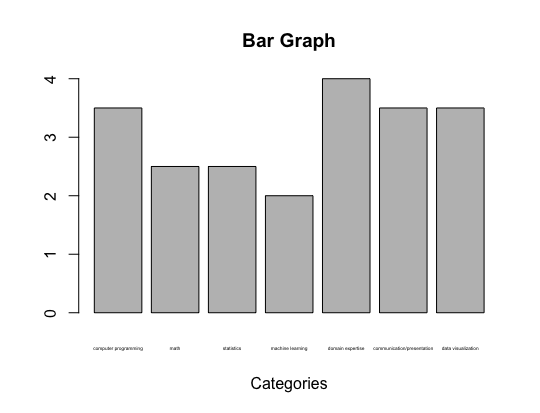
spencer = data.frame(categories, rankings)

spencer

barplot(rankings, names.arg = categories, xlab='Categories',

main = 'Bar Graph', cex.names=0.3)





Note: I could not figure out how to fit all the names without making them very small.