Assignment 0

Justin Sallese

1/29/2017

GitHub Username:jsallese

[GitHub URL](https://github.com/jsallese/SRT411-Assignment-0)

[A very Short introduction to R](https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf)

The overall purpose of this assignment is to increase ones knowladge of the r programming language by doing the exercises found in the document "A (very) short introduction to R" by Paul Torfs & Claudia Brauer. Below can be found the code used to complete the to do exercises found within the introduction to R 3.1

((2014-2014)/(2014-1995))\*100

## [1] 0

3.2

a <- 2014-2014  
b <- 2014-1995  
c <- 100  
d <- (a/b)\*c  
d

## [1] 0

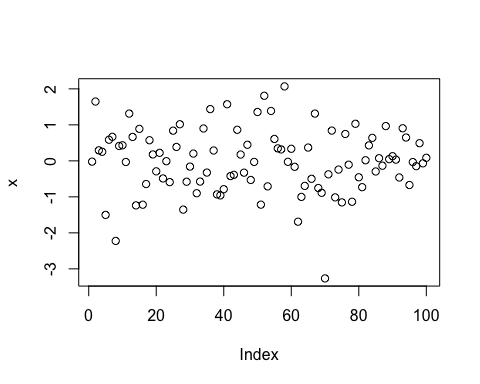
3.4

z <- c(4,5,8,11)  
sum(z)

## [1] 28

3.5

x = rnorm(100)  
plot(x)

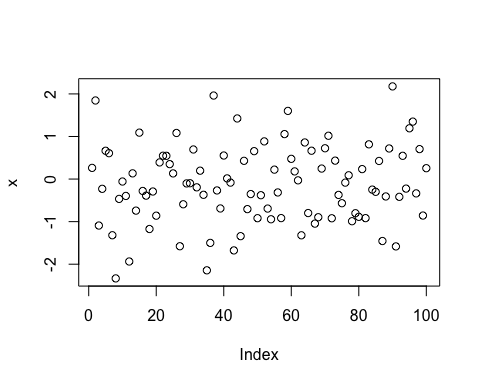


4

help(sqrt)

5

source("firstscript.R")

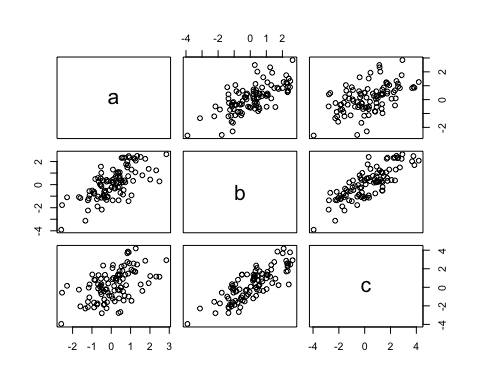
 6.2

P <- seq(from=31, to=60, by=1)  
Q <- matrix(P, ncol=5, nrow=6)  
Q

## [,1] [,2] [,3] [,4] [,5]  
## [1,] 31 37 43 49 55  
## [2,] 32 38 44 50 56  
## [3,] 33 39 45 51 57  
## [4,] 34 40 46 52 58  
## [5,] 35 41 47 53 59  
## [6,] 36 42 48 54 60

6.3

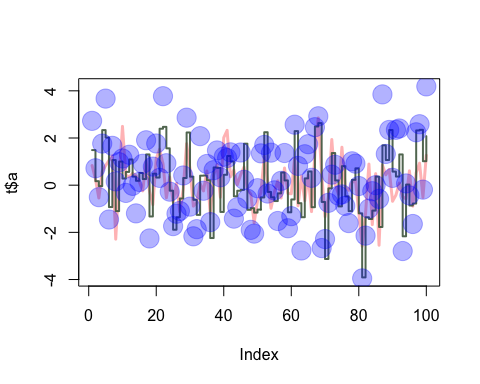
x1 <- rnorm(100)  
x2 <- rnorm(100)  
x3 <- rnorm(100)  
t <- data.frame(a= x1, b= x1+x2, c= x1+x2+x3)  
plot(t)



The results of the script run shows the plotting of 3 variables denoting 100 random numbers each. For each variable in the data frame are added to each other as the variables continue. The plot shows the result of the random number variable being added together.

7

plot(t$a, type="l", ylim=range(t),  
 lwd=3, col=rgb(1,0,0,0.3))  
lines(t$b, type="s", lwd=2,  
 col=rgb(0.3,0.4,0.3,0.9))  
points(t$c, pch=20, cex=4,  
 col=rgb(0,0,1,0.3))



The meainging of the rgb variable denotes the color of certain aspects of the plot that is designated. The lwd variable denotes the width of the line within the plot. The variable pch denotes the type of symbol to use within the plot. The variable cex denotes the size of the symbols used within the plot.

8

d = read.table(file="tst1.txt", header = TRUE)  
d$g <- d$g \* 5  
write.table(d, file = "tst2.txt", row.names = FALSE)

9

v <- rnorm(100)  
v2 <- sqrt(v)

## Warning in sqrt(v): NaNs produced

mean(v2)

## [1] NaN

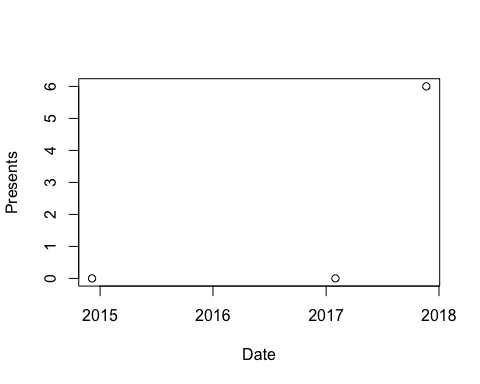
v2

## [1] NaN 0.79604815 NaN NaN 1.63022723 0.82604088  
## [7] NaN 0.53322537 1.06921076 NaN NaN 1.17312264  
## [13] NaN 0.79068566 NaN 1.22159556 0.07263297 0.42798085  
## [19] NaN NaN NaN 0.67505173 0.71586953 NaN  
## [25] NaN NaN NaN NaN 1.00831324 NaN  
## [31] 1.09516529 NaN NaN NaN NaN 0.93766923  
## [37] NaN NaN 1.25428949 0.92318965 1.37827947 NaN  
## [43] 1.01429559 NaN 0.32083261 NaN NaN NaN  
## [49] NaN 0.33659065 NaN NaN NaN 0.53853988  
## [55] 1.12968124 NaN NaN 1.38617447 NaN 0.85386678  
## [61] 0.63967016 1.04277920 NaN 1.00212696 NaN 0.34609914  
## [67] 0.61807547 NaN NaN 0.79998146 NaN NaN  
## [73] 1.07249020 0.61906032 0.64294818 NaN 0.42843523 NaN  
## [79] NaN NaN 1.15431163 NaN 0.37633998 0.75419461  
## [85] NaN NaN 0.70654252 0.36140762 NaN NaN  
## [91] 0.95758039 0.76067808 NaN NaN 1.43315777 NaN  
## [97] 1.07489145 NaN 0.09975222 NaN

The results display all the result whether or not they can be calculated. If they can be calculated then they are displayed, if the numbers cannot be calculated then the results are displayed as NaN.

10.2

Date <- strptime( c("20170131","20141206","20171121"),format="%Y%m%d")  
Presents <- c(0,0,6)  
plot(Date,Presents)



11.2

num <- seq(from = 1, to = 100, by=1)  
f=c()  
for(l in 1:100)  
{  
if(num[l]<5 | num[l]>90)  
 {f[l]=num[l]\*10}  
else  
 {f[l]=num[l]\*0.1}  
}  
f

## [1] 10.0 20.0 30.0 40.0 0.5 0.6 0.7 0.8 0.9 1.0  
## [11] 1.1 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2.0  
## [21] 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.9 3.0  
## [31] 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 4.0  
## [41] 4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.8 4.9 5.0  
## [51] 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9 6.0  
## [61] 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8 6.9 7.0  
## [71] 7.1 7.2 7.3 7.4 7.5 7.6 7.7 7.8 7.9 8.0  
## [81] 8.1 8.2 8.3 8.4 8.5 8.6 8.7 8.8 8.9 9.0  
## [91] 910.0 920.0 930.0 940.0 950.0 960.0 970.0 980.0 990.0 1000.0

11.3

fun1 = function(arg1)  
{len = length(arg1)  
 for(q in 1:len)  
{if (arg1[q] < 5 | arg1[q] > 90)  
{arg1[q] = arg1[q] \* 10}  
else  
{arg1[q] = arg1[q] \* 0.1}}  
 return (arg1)  
}  
m=4:32  
fun1(arg1=m)

## [1] 40.0 0.5 0.6 0.7 0.8 0.9 1.0 1.1 1.2 1.3 1.4 1.5 1.6 1.7  
## [15] 1.8 1.9 2.0 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.9 3.0 3.1  
## [29] 3.2

11.3 Footer

fun1 = function(arg1)  
{  
{if (arg1 < 5 | arg1 > 90)  
{arg1 = arg1 \* 10}  
ifelse  
{arg1 = arg1 \* 0.1}}  
 return (arg1)  
}  
m=1:100  
fun1(arg1=m)

## Warning in if (arg1 < 5 | arg1 > 90) {: the condition has length > 1 and  
## only the first element will be used

## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17  
## [18] 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34  
## [35] 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51  
## [52] 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68  
## [69] 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85  
## [86] 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

Work Cited

Broman, Karl. "Knitr with R Markdown." Sitewide ATOM. N.p., n.d. Web. 01 Feb. 2017.

Broman, Karl. "Markdown." Sitewide ATOM. N.p., n.d. Web. 01 Feb. 2017.

Falster, Rich FitzJohn & Daniel. "Nice R Code." Writing Reproducible Reports in R with Markdown, Knitr and Pandoc - Nice R Code. N.p., n.d. Web. 01 Feb. 2017.

R Markdown Cheat Sheet. Digital image. RStudio. N.p., Aug. 2014. Web. 1 Feb. 2017.

"R Markdown." R Markdown. N.p., n.d. Web. 01 Feb. 2017.

Torfs, Paul, and Claudia Brauer. "A (very) Short Introduction to R." A (very) Short Introduction to R (n.d.): 1-12. Cran.r-preogect.org. Rstudio, 3 Mar. 2014. Web. 1 Feb. 2017.