Assignment0

Maggie Yu

February 14, 2018

Github user: myu31 Github URL: https://github.com/myu31/SRT411 This assignment is from "https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf" The document is a introduction to R language and the To Do's are exercises that can be done to try basic R Language.

TO DO's: 1)Compute the difference between 2014 and the year you started at this university and divide this by the di???erence between 2014 and the year you were born. Multiply this with 100 to get the percentage of your life you have spent at this university. Use brackets if you need them.

```
((2018-2017)/(2018-1991))*100

## [1] 3.703704
```

2)Repeat the previous ToDo, but with several steps in between. You can give the variables any name you want, but the name has to start with a letter.

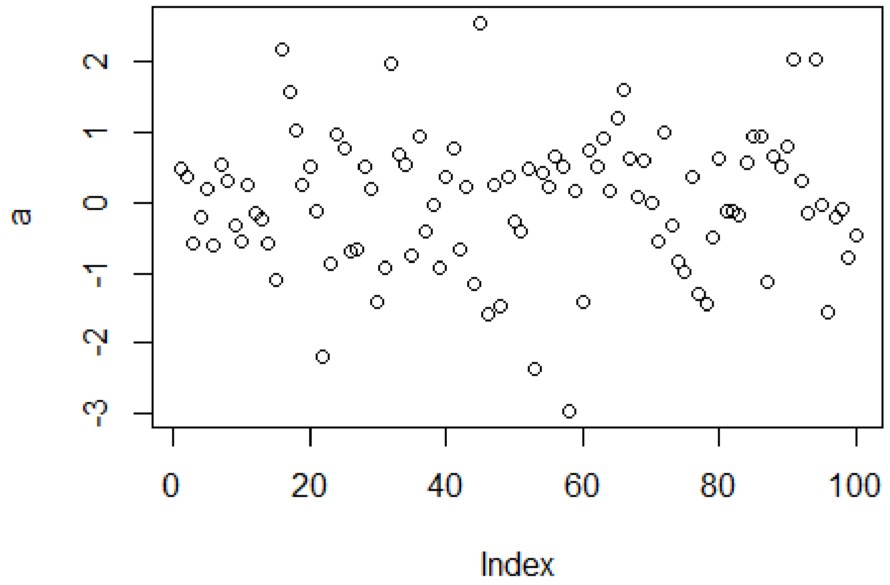
```
a = 2018
b = 2017
c = 1991
d = ((a-b)/(a-c))*100
d

## [1] 3.703704
```

3)Compute the sum of 4, 5, 8 and 11 by first combining them into a vector and then using the function sum.

```
a=c(4,5,8,11)
sum(a)

## [1] 28
```

4)Plot 100 normal random numbers.

```
a= rnorm(100)
plot(a)
```



5)Find help for the sqrt function.

```
help(sqrt)

## starting httpd help server ... done
```

6)Make a file called firstscript.R containing Rcode that generates 100 random numbers and plots them, and run this script several times.

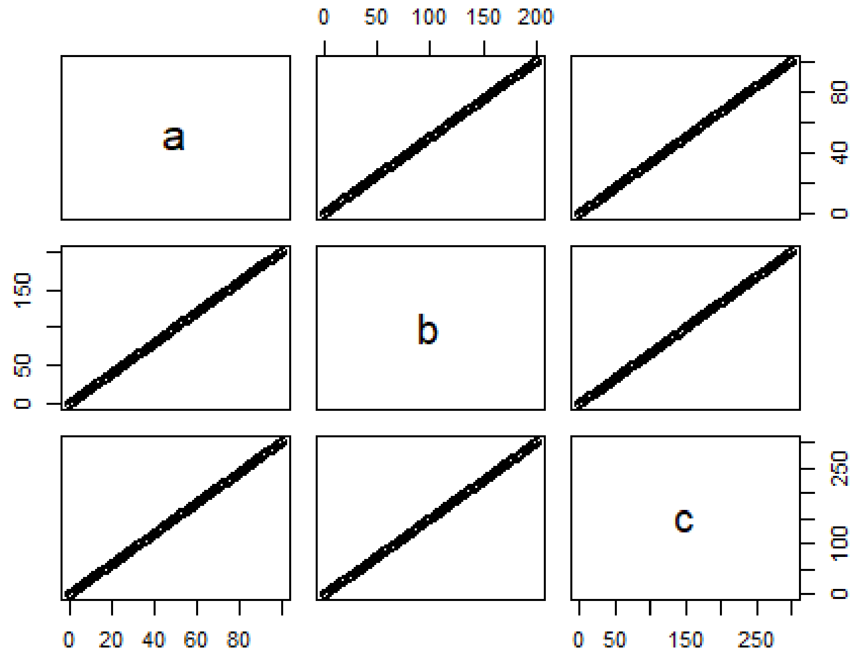
```
write(rnorm(100),file ="firstscript.R")
```

7)Put the numbers 31 to 60 in a vector named P and in a matrix with 6 rows and 5 columns named Q. Tip: use the function seq. Look at the different ways scalars, vectors and matrices are denoted in the workspace window.

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

##  [1] 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53
## [24] 54 55 56 57 58 59 60

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
```

8)Make a script file which constructs three random normal vectors of length 100. Call these vectors x1, x2 and x3. Make a data frame called t with three columns (called a, b and c) containing respectively x1, x1+x2 and x1+x2+x3. Call the following functions for this data frame: plot(t) and sd(t). Can you understand the results? Rerun this script a few times.

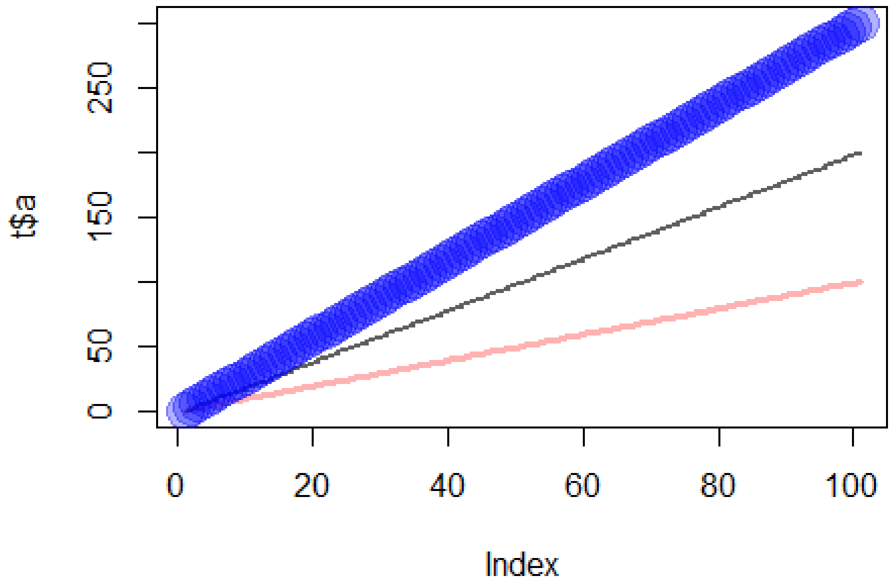
```
x1=seq(from=0, to=100,by=1)
x2=seq(from=0, to=100, by=1)
x3=seq(from=0, to=100, by=1)
t=data.frame(a=x1,b=x1+x2,c=x1+x2+x3)
plot(t)
```



The graph shows the graph of a which is x1, graph of b which is x1+x2 and graph of c which is x1+x2+x3

9)Add these lines to the script file of the previous section. Try to find out, either by experimenting or by using the help, what the meaning is of rgb, the last argument of rgb, lwd, pch, cex.

```
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(0.3,0.3,0.3,0.9))
points(t$c,pch=20,cex=4,col=rgb(0,0,1,0.3))
```



10)Make a file called tst1.txt in Notepad from the example in Figure 4 and store it in your working directory. Write a script to read it, to multiply the column called g by 5 and to store it as tst2.txt

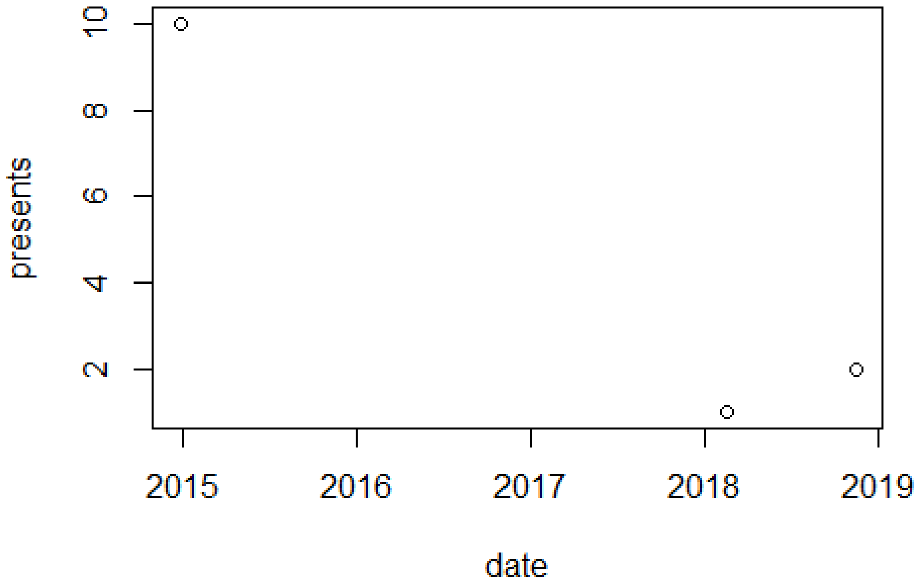
```
d=data.frame(g=c(3,4,5), h=c(12,43,54))
d

##   g  h
## 1 3 12
## 2 4 43
## 3 5 54

write.table(d,file="tst2.txt",row.names=FALSE)
d2=read.table(file="tst2.txt",header=TRUE)
d2$g*5

## [1] 15 20 25
```

11)Compute the mean of the square root of a vector of 100 random numbers. What happens?

```
sqrt(mean(rnorm(100)))

## Warning in sqrt(mean(rnorm(100))): NaNs produced

## [1] NaN
```

12)Make a graph with on the x-axis: today, Sinterklaas 2014 and your next birthday and on the y-axis the number of presents you expect on each of these days. Tip: make two vectors first.

```
date1=strptime(c("20180215","20141225","20181115"),format="%Y%m%d")
date=date1
presents=c(1,10,2)
plot(date,presents)
```



13)Make a vector from 1 to 100. Make a for-loop which runs through the whole vector. Multiply the elements which are smaller than 5 and larger than 90 with 10 and the other elements with 0.1.

```
a=seq(from=1, to=100, by=1)
for(i in 1:100)
{
  if(a[i]<5|a[i]>90)
  {
    a[i]=a[i]*10
  }
  else
  {
    a[i]=a[i]*0.1
  }
}
a
```

```
##   [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
```

14)Write a function for the previous ToDo, so that you can feed it any vector you like (as argument). Use a for-loop in the function to do the computation with each element. Use the standard R function length in the specification of the counter. a

```
fun1=function(arg1)
{
  b=length(arg1)
  for(i in 1:b)
  {
      if(arg1[i]<5 | arg1[i]>90)
      {
        arg1[i]=arg1[i]*10
      }
      else
      {
        arg1[i]=arg1[i]*0.1
      }
  }
  return(arg1)
}
a=1:100
fun1(arg1=a)
```

```
##   [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
```

15)The ToDo above can be done more easily and quickly without a for-loop but with regular vectorcomputations.

```
fun1=function(arg1)
{
  ifelse(arg1<5|arg1>90,arg1*10,arg1*0.1)
}
x=seq(1,100,by=1)
fun1(x)

##   [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
```