

hwk_3

Andrew Mikolinski

2023-10-11

Question 1

A = matrix(data = runif(n = 1:500), nrow = 50, ncol = 10) colnames(A) = paste("lake", 1:10, sep = "_")
set.seed(12) # to be reproducible

```
set.seed(12)

A = matrix(data = runif(n = 1:500), nrow = 50, ncol = 10)

colnames(A) = paste("lake", 1:10, sep = "_")

column_means <- vector(mode = "numeric", length = ncol(A))

for (i in 1:ncol(A)) {

  column_means[i] <- mean(A[[i]])
}

names(column_means) <- colnames(A)

column_means
```

```
##      lake_1      lake_2      lake_3      lake_4      lake_5      lake_6
## 0.069360916 0.817775199 0.942621732 0.269381876 0.169348123 0.033895622
##      lake_7      lake_8      lake_9      lake_10
## 0.178785004 0.641665366 0.022877743 0.008324827
```

```
y2<-numeric()

for (i in c(1:10)){

  y1<-mean(A[,i])

  y2<-c(y2,y1)

}

names(y2)<-colnames(A)

y2
```

```
## lake_1 lake_2 lake_3 lake_4 lake_5 lake_6 lake_7 lake_8
## 0.4601492 0.4992815 0.5987037 0.4580486 0.4719578 0.4965216 0.5110536 0.4577936
## lake_9 lake_10
## 0.5193423 0.4856413
```

Question 2

```
x = matrix(c(1:27), nrow = 3, ncol=9)
x
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## [1,]    1    4    7   10   13   16   19   22   25
## [2,]    2    5    8   11   14   17   20   23   26
## [3,]    3    6    9   12   15   18   21   24   27
```

```
for(i in ncol(x[,1:2])){
  X = c(x[,1:3 ])
  Y = c(x[,4:6 ])
  Z = c(x[,7:9 ])
  print(matrix(nrow=3, ncol=3, data=paste("",X," ",Y," ",Z,"")), collapse = " ")
}
```

```
##      [,1]      [,2]      [,3]
## [1,] " 1 , 10 , 19 " " 4 , 13 , 22 " " 7 , 16 , 25 "
## [2,] " 2 , 11 , 20 " " 5 , 14 , 23 " " 8 , 17 , 26 "
## [3,] " 3 , 12 , 21 " " 6 , 15 , 24 " " 9 , 18 , 27 "
```

Question 3

```
Fibonacci <- numeric(30)
Fibonacci[1] <- Fibonacci[2] <- 1
for (i in 3:30)
  Fibonacci[i] <- Fibonacci[i - 2] + Fibonacci[i - 1]
print(Fibonacci)
```

```
## [1]      1      1      2      3      5      8     13     21     34     55
## [11]     89    144    233    377    610    987   1597   2584   4181   6765
## [21]   10946   17711   28657   46368   75025  121393  196418  317811  514229  832040
```

Question 4&5

```
library("stringr")
top105 = readLines("http://www.textfiles.com/music/ktop100.txt")
top105 = top105[-c(1:10, 64, 65)]

top1052 <- str_extract_all(top105, pattern = "[[:alpha:]]-?&,( )' ]", simplify = FALSE)
```

```

top1052[c(107)] <- top1052[c(107)-.3]
top1052[c(98)] <- top1052[c(98) - 0.69]
top1052[c(29)] <- top1052[c(29)-0.3]
top1052[c(16)] <- top1052[c(16)-0.2]
top1052[c(9)] <- top1052[c(9)-0.3]
top1052 <- top1052[-c(108:111)]
top1052

```

```

## [[1]]
## [1] "1" "."
##
## [[2]]
## [1] "2" "."
##
## [[3]]
## [1] "3" "." "." "." "."
##
## [[4]]
## [1] "4" "."
##
## [[5]]
## [1] "5" "." "." "." "."
##
## [[6]]
## [1] "6" "."
##
## [[7]]
## [1] "7" "."
##
## [[8]]
## [1] "8" "."
##
## [[9]]
## [1] "8" "."
##
## [[10]]
## [1] "1" "0" "."
##
## [[11]]
## [1] "1" "1" "."
##
## [[12]]
## [1] "1" "2" "."
##
## [[13]]
## [1] "1" "3" "."
##
## [[14]]
## [1] "1" "4" "."
##
## [[15]]
## [1] "1" "5" "."
##

```

```

## [[16]]
## [1] "1" "5" "."
##
## [[17]]
## [1] "1" "7" "."
##
## [[18]]
## [1] "1" "8" "."
##
## [[19]]
## [1] "1" "9" "."
##
## [[20]]
## [1] "2" "0" "." "." "." "."
##
## [[21]]
## [1] "2" "1" "." "." "." "."
##
## [[22]]
## [1] "2" "2" "."
##
## [[23]]
## [1] "2" "3" "."
##
## [[24]]
## [1] "2" "4" "."
##
## [[25]]
## [1] "2" "5" "."
##
## [[26]]
## [1] "2" "6" "."
##
## [[27]]
## [1] "2" "7" "."
##
## [[28]]
## [1] "2" "8" "."
##
## [[29]]
## [1] "2" "8" "."
##
## [[30]]
## [1] "3" "0" "."
##
## [[31]]
## [1] "3" "1" "."
##
## [[32]]
## [1] "3" "2" "." "." "." "."
##
## [[33]]
## [1] "3" "3" "."
##

```

```

## [[34]]
## [1] "3" "4" "."
##
## [[35]]
## [1] "3" "5" "."
##
## [[36]]
## [1] "3" "6" "."
##
## [[37]]
## [1] "3" "7" "."
##
## [[38]]
## [1] "3" "8" "."
##
## [[39]]
## [1] "3" "9" "."
##
## [[40]]
## [1] "4" "0" "."
##
## [[41]]
## [1] "4" "1" "." "." "." "."
##
## [[42]]
## [1] "4" "2" "."
##
## [[43]]
## [1] "4" "3" "."
##
## [[44]]
## [1] "4" "4" "."
##
## [[45]]
## [1] "4" "5" "."
##
## [[46]]
## [1] "4" "6" "."
##
## [[47]]
## [1] "4" "7" "."
##
## [[48]]
## [1] "4" "8" "."
##
## [[49]]
## [1] "4" "9" "."
##
## [[50]]
## [1] "5" "0" "."
##
## [[51]]
## [1] "5" "1" "."
##

```

```

## [[52]]
## [1] "5" "2" "."
##
## [[53]]
## [1] "5" "3" "." "."
##
## [[54]]
## [1] "5" "6" "."
##
## [[55]]
## [1] "5" "7" "."
##
## [[56]]
## [1] "5" "8" "."
##
## [[57]]
## [1] "5" "9" "."
##
## [[58]]
## [1] "6" "0" "."
##
## [[59]]
## [1] "6" "1" "."
##
## [[60]]
## [1] "6" "2" "."
##
## [[61]]
## [1] "6" "3" "."
##
## [[62]]
## [1] "6" "4" "."
##
## [[63]]
## [1] "6" "5" "."
##
## [[64]]
## [1] "6" "6" "." "." "." "." "." "."
##
## [[65]]
## [1] "6" "7" "."
##
## [[66]]
## [1] "6" "8" "."
##
## [[67]]
## [1] "6" "9" "."
##
## [[68]]
## [1] "7" "0" "."
##
## [[69]]
## [1] "7" "1" "."
##

```

```

## [[70]]
## [1] "7" "2" "." "." "." "."
##
## [[71]]
## [1] "7" "3" "."
##
## [[72]]
## [1] "7" "4" "."
##
## [[73]]
## [1] "7" "5" "."
##
## [[74]]
## [1] "7" "6" "."
##
## [[75]]
## [1] "7" "7" "."
##
## [[76]]
## [1] "7" "8" "." "." "." "." "." "." "."
##
## [[77]]
## [1] "7" "9" "."
##
## [[78]]
## [1] "8" "0" "." "."
##
## [[79]]
## [1] "8" "1" "."
##
## [[80]]
## [1] "8" "2" "."
##
## [[81]]
## [1] "8" "3" "."
##
## [[82]]
## [1] "8" "3" "."
##
## [[83]]
## [1] "8" "4" "."
##
## [[84]]
## [1] "8" "5" "."
##
## [[85]]
## [1] "8" "6" "."
##
## [[86]]
## [1] "8" "7" "."
##
## [[87]]
## [1] "8" "8" "."
##

```

```

## [[88]]
## [1] "8" "9" "."
##
## [[89]]
## [1] "9" "0" "."
##
## [[90]]
## [1] "9" "1" "."
##
## [[91]]
## [1] "9" "1" "."
##
## [[92]]
## [1] "9" "2" "."
##
## [[93]]
## [1] "9" "3" "."
##
## [[94]]
## [1] "9" "4" "."
##
## [[95]]
## [1] "9" "5" "."
##
## [[96]]
## [1] "9" "6" "." "." "."
##
## [[97]]
## [1] "9" "7" "."
##
## [[98]]
## [1] "9" "7" "."
##
## [[99]]
## [1] "9" "8" "."
##
## [[100]]
## [1] "9" "9" "."
##
## [[101]]
## [1] "1" "0" "0" "."
##
## [[102]]
## [1] "1" "0" "1" "."
##
## [[103]]
## [1] "1" "0" "2" "." "." "." "."
##
## [[104]]
## [1] "1" "0" "3" "."
##
## [[105]]
## [1] "1" "0" "4" "."
##

```



```
## [[106]]
## [1] "1" "0" "5" "."
##
## [[107]]
## [1] "1" "0" "5" "."
```

```
top105 = readLines("http://www.textfiles.com/music/ktop100.txt")
top105 = top105[-c(1:10, 64, 65)]
top1053 <- str_extract_all(top105, pattern = "[^\\.[[:alpha:]]-?&,( )' ]", simplify = FALSE)
top1053 <- top1053[-c(108:111)]
top1053[c(107)] <- top1053[c(107)]-.3]
top1053[c(98)] <- top1053[c(98)] - 0.69]
top1053[c(29)] <- top1053[c(29)]-0.3]
top1053[c(16)] <- top1053[c(16)]-0.2]
top1053[c(9)] <- top1053[c(9)]-0.3]
duplicated(top1053)
```

```
## [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE
## [13] FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [25] FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE
## [85] FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE
## [97] FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE
```