lab09

sl

2022-10-26

## Problem 2.

Create a n x k dataset with all its entries distributed poission with mean lambda

set.seed(1235)  
fun1 <- function(n = 100, k = 4, lambda = 4) {  
 x <- NULL  
   
 for (i in 1:n)  
 x <- rbind(x, rpois(k, lambda))  
   
 return(x)  
}  
f1 <- fun1(1000,4)  
mean(f1)

## [1] 4.03725

#f1 <- fun1(10000,4)  
#f1 <- fun1(50000,4)  
  
fun1alt <- function(n = 100, k = 4, lambda = 4) {  
 # YOUR CODE HERE  
   
 x <- matrix( rpois(n\*k, lambda), ncol = 4)  
   
 return(x)  
}  
f1 <- fun1alt(50000,4)  
  
  
# Benchmarking  
microbenchmark::microbenchmark(  
 fun1(),  
 fun1alt()  
)

## Unit: microseconds  
## expr min lq mean median uq max neval  
## fun1() 719.787 1134.6810 2506.417 1374.1195 2034.065 22574.48 100  
## fun1alt() 31.730 41.0495 1302.184 48.3355 60.187 119913.45 100

d <- matrix(1:16, ncol = 4)  
d

## [,1] [,2] [,3] [,4]  
## [1,] 1 5 9 13  
## [2,] 2 6 10 14  
## [3,] 3 7 11 15  
## [4,] 4 8 12 16

diag(d)

## [1] 1 6 11 16

d[2,1]

## [1] 2

d[c(1,6,11,16)]

## [1] 1 6 11 16

cbind(1:4,1:4)

## [,1] [,2]  
## [1,] 1 1  
## [2,] 2 2  
## [3,] 3 3  
## [4,] 4 4

d[cbind(1:4,1:4)]

## [1] 1 6 11 16

## Problem 3.

Find the column max (hint: Checkout the function max.col()).

# Data Generating Process (10 x 10,000 matrix)  
set.seed(1234)  
M <- matrix(runif(12), ncol = 4)  
M

## [,1] [,2] [,3] [,4]  
## [1,] 0.1137034 0.6233794 0.009495756 0.5142511  
## [2,] 0.6222994 0.8609154 0.232550506 0.6935913  
## [3,] 0.6092747 0.6403106 0.666083758 0.5449748

# Find each column's max value  
fun2 <- function(x) {  
 apply(x, 2, max)  
}  
fun2(x=M)

## [1] 0.6222994 0.8609154 0.6660838 0.6935913

fun2alt <- function(x) {  
 # YOUR CODE HERE  
 idx <- max.col(t(x))  
 x[cbind(idx,1:4)]  
}  
fun2alt(x=M)

## [1] 0.6222994 0.8609154 0.6660838 0.6935913

x <- matrix(rnorm(1e4), nrow=10)  
  
# Benchmarking  
microbenchmark::microbenchmark(  
 fun2(x),  
 fun2alt(x)  
)

## Unit: microseconds  
## expr min lq mean median uq max neval  
## fun2(x) 2705.372 3658.2195 7290.209 5063.3490 8324.5430 33789.364 100  
## fun2alt(x) 227.009 293.1955 617.993 354.4535 473.4385 8041.965 100