novikovPrac2.R

Новиков Денис Владиславович ИНБО-05-22

2024-02-13

# Часть 1 -----------------------------------------------------------------  
  
# Задание 1 ---------------------------------------------------------------  
  
A <- matrix(3, 3, 4)  
A

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

A[1,3] <- 4  
A[2,1] <- 1  
A[3,2] <- NA  
A[3,4] <- 1  
A

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

# Задание 2 ---------------------------------------------------------------  
  
a <- c(1, 3, 4, 9, NA)  
b <- c(5, 6, 7, 0, 2)  
c <- c(9, 10, 13, 1, 20)  
  
B <- matrix(c(a, b, c), nrow = 5, byrow = T, dimnames = list(c("A1", "A2", "A3", "A4", "A5"), c("B1", "B2", "B3")))  
B

## B1 B2 B3  
## A1 1 3 4  
## A2 9 NA 5  
## A3 6 7 0  
## A4 2 9 10  
## A5 13 1 20

C <- matrix(c(a, b, c), nrow = 3, byrow = T, dimnames = list(c("row1", "row2", "row3"), c("col1", "col2", "col3", "col4", "col5")))  
C

## col1 col2 col3 col4 col5  
## row1 1 3 4 9 NA  
## row2 5 6 7 0 2  
## row3 9 10 13 1 20

# Задание 3 ---------------------------------------------------------------  
  
names <- c("Jane", "Michael", "Mary", "George")  
ages <- c(8, 6, 28, 45)  
gender <- c(0, 1, 0, 1)  
  
D <- cbind(names, ages, gender)  
D

## names ages gender  
## [1,] "Jane" "8" "0"   
## [2,] "Michael" "6" "1"   
## [3,] "Mary" "28" "0"   
## [4,] "George" "45" "1"

ages\_sq <- ages^2  
D <- cbind(D, ages\_sq)  
D

## names ages gender ages\_sq  
## [1,] "Jane" "8" "0" "64"   
## [2,] "Michael" "6" "1" "36"   
## [3,] "Mary" "28" "0" "784"   
## [4,] "George" "45" "1" "2025"

D[,1]

## [1] "Jane" "Michael" "Mary" "George"

as.integer(D[, 2])

## [1] 8 6 28 45

as.integer(D[, 3])

## [1] 0 1 0 1

as.integer(D[, 4])

## [1] 64 36 784 2025

# Задание 4 ---------------------------------------------------------------  
  
info <- list(names = names, ages = ages, gender = gender)  
info$names[info$names == "Michael"]

## [1] "Michael"

info$gender

## [1] 0 1 0 1

names(info) <- c("name", "age", "gender")  
info$name

## [1] "Jane" "Michael" "Mary" "George"

drinks <- c("juice", "tea", "rum", "coffee")  
info$drinks <- drinks  
new <- list(names = "John", age = 2, gender = 1, drinks = "milk")  
info$name[5] <- new$name  
info$age[5] <- new$age  
info$gender[5] <- new$gender  
info$drinks[5] <- new$drinks  
info

## $name  
## [1] "Jane" "Michael" "Mary" "George" "John"   
##   
## $age  
## [1] 8 6 28 45 2  
##   
## $gender  
## [1] 0 1 0 1 1  
##   
## $drinks  
## [1] "juice" "tea" "rum" "coffee" "milk"

# Задание 5 ---------------------------------------------------------------  
  
s <- "a,b,c,d"  
let <- strsplit(s, ",")  
class(let)

## [1] "list"

s <- unlist(let)  
s

## [1] "a" "b" "c" "d"

index <- "0,72;0,38;0,99;0,81;0,15;0,22;0,16;0,4;0,24"  
temp <- strsplit(index, ";")  
temp

## [[1]]  
## [1] "0,72" "0,38" "0,99" "0,81" "0,15" "0,22" "0,16" "0,4" "0,24"

class(temp)

## [1] "list"

temp <- sub(",", ".", unlist(temp))  
temp

## [1] "0.72" "0.38" "0.99" "0.81" "0.15" "0.22" "0.16" "0.4" "0.24"

I <- as.numeric(temp)  
I

## [1] 0.72 0.38 0.99 0.81 0.15 0.22 0.16 0.40 0.24

# Часть 2 -----------------------------------------------------------------  
  
# Задание 1 ---------------------------------------------------------------  
  
A <- diag(c(4, 9), 2, 2)  
rownames(A) <- c("eq1", "eq2")  
colnames(A) <- c("x1", "x2")  
A

## x1 x2  
## eq1 4 0  
## eq2 0 9

# Задание 2 ---------------------------------------------------------------  
  
e <- eigen(A)  
e$values

## [1] 9 4

# собственные значения равны 9 и 4  
# так как являются решением уравнения(4-λ)\*(9-λ) = 0  
  
# Задание 3 ---------------------------------------------------------------  
  
I <- diag(2)  
I

## [,1] [,2]  
## [1,] 1 0  
## [2,] 0 1

B <- I - A  
B

## x1 x2  
## eq1 -3 0  
## eq2 0 -8

# Задание 4 ---------------------------------------------------------------  
  
f <- c(4, 2)  
f

## [1] 4 2

u <- c(0.2, -0.3)  
u

## [1] 0.2 -0.3

# Задание 5 ---------------------------------------------------------------  
  
u\_result <- solve(A, f)  
u\_result

## x1 x2   
## 1.0000000 0.2222222

# Задание 6 ---------------------------------------------------------------  
  
results <- matrix(0, nrow = 2, ncol = 7)  
  
for (i in 1:7) {  
 u <- B %\*% u + f  
 results[, i] <- u  
}  
  
results

## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 3.4 -6.2 22.6 -63.8 195.4 -582.2 1750.6  
## [2,] 4.4 -33.2 267.6 -2138.8 17112.4 -136897.2 1095179.6

# Задание 7 ---------------------------------------------------------------  
  
difference <- abs(results[, 7] - u\_result)  
difference

## x1 x2   
## 1749.6 1095179.4

# Задание 8 ---------------------------------------------------------------  
  
max\_val <- max(A)  
max\_val

## [1] 9

A

## x1 x2  
## eq1 4 0  
## eq2 0 9

A <- A / max\_val  
A

## x1 x2  
## eq1 0.4444444 0  
## eq2 0.0000000 1

f

## [1] 4 2

f <- f / max\_val  
f

## [1] 0.4444444 0.2222222

# Задание 9 ---------------------------------------------------------------  
  
u2 <- c(0.2, -0.3)  
  
e2 <- eigen(A)  
e2$values

## [1] 1.0000000 0.4444444

I2 <- diag(2)  
I2

## [,1] [,2]  
## [1,] 1 0  
## [2,] 0 1

B2 <- A - I  
B2

## x1 x2  
## eq1 -0.5555556 0  
## eq2 0.0000000 0

u\_result2 <- solve(A, f)  
u\_result2

## x1 x2   
## 1.0000000 0.2222222

results2 <- matrix(0, nrow = 2, ncol = 7)  
  
for (i in 1:7) {  
 u2 <- B2 %\*% u2 + f  
 results2[, i] <- u2  
}  
  
results2

## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 0.3333333 0.2592593 0.3004115 0.2775492 0.2902505 0.2831942 0.2871143  
## [2,] 0.2222222 0.2222222 0.2222222 0.2222222 0.2222222 0.2222222 0.2222222

difference2 <- abs(results2[, 7] - u\_result2)  
difference2

## x1 x2   
## 0.7128857 0.0000000

# Задание 10 --------------------------------------------------------------  
  
difference3 <- abs(difference - difference2)  
difference3

## x1 x2   
## 1748.887 1095179.378

# Часть 3 -----------------------------------------------------------------  
  
step <- 1 # Шаг сетки  
dekart\_begin <- -5 # Начало сетки  
dekart\_end <- 5 # Конец сетки  
  
# Задание сеточной поверхности  
x <- seq(from = dekart\_begin, to = dekart\_end, by = step)  
y <- x  
  
# Задание двумерной функции на координатной сетке  
surface\_matrix <- outer(X = x,  
 Y = y,  
 FUN = function(x,y) Re(exp(-1i \* 0.5 \* x \* y)))  
dimnames(surface\_matrix) <- list(x, y)  
  
# Задание 1 ---------------------------------------------------------------  
  
# Количество элементов матрицы  
write(paste("number of matrix elements:", length(surface\_matrix)), "summary.txt")  
  
file\_conn <- file("summary.txt", "a")  
  
# Размерность строк и столбцов  
writeLines(paste("number of rows:", nrow(surface\_matrix)), file\_conn)  
writeLines(paste("number of cols:", ncol(surface\_matrix)), file\_conn)  
  
# Сумма элементов главной диагонали  
writeLines(paste("sum of main diag elements:", sum(diag(surface\_matrix))), file\_conn)  
  
# Сумма элементов серединного среза по строкам  
writeLines(paste("sum of middle row elements:", sum(surface\_matrix[nrow(surface\_matrix) %/% 2, ])), file\_conn)  
  
# Сумма элементов серединного среза по столбцам  
writeLines(paste("sum of middle column elements:", sum(surface\_matrix[, ncol(surface\_matrix) %/% 2])), file\_conn)  
  
# Суммы строк матрицы  
writeLines(paste("row sums:", rowSums(surface\_matrix)), file\_conn)  
  
# Суммы столбцов матрицы  
writeLines(paste("col sums:", colSums(surface\_matrix)), file\_conn)  
  
close(file\_conn)  
  
# Задание 2 ---------------------------------------------------------------  
  
# Для генерации отчёта закомментировал эти строки  
# step2 <- as.numeric(readline(prompt = "Введите шаг сетки: ")) # Шаг сетки  
# dekart\_begin2 <- as.numeric(readline(prompt = "Введите начало сетки: ")) # Начало сетки  
# dekart\_end2 <- as.numeric(readline(prompt = "Введите конец сетки: ")) # Конец сетки  
  
# Введём пример данных, которые может ввести пользователь:  
step2 <- 2 # Шаг сетки  
dekart\_begin2 <- -18 # Начало сетки  
dekart\_end2 <- 18 # Конец сетки  
  
# Задание сеточной поверхности  
x2 <- seq(from = dekart\_begin2, to = dekart\_end2, by = step2)  
y2 <- x2  
  
# Задание двумерной функции на координатной сетке  
surface\_matrix2 <- outer(X = x2,  
 Y = y2,  
 FUN = function(x2,y2) Re(exp(-1i \* 0.5 \* x2 \* y2)))  
dimnames(surface\_matrix2) <- list(x2, y2)  
  
# Количество элементов матрицы  
write(paste("number of matrix elements:", length(surface\_matrix2)), "summary2.txt")  
  
file\_conn <- file("summary2.txt", "a")  
  
# Размерность строк и столбцов  
writeLines(paste("number of rows:", nrow(surface\_matrix2)), file\_conn)  
writeLines(paste("number of cols:", ncol(surface\_matrix2)), file\_conn)  
  
# Сумма элементов главной диагонали  
writeLines(paste("sum of main diag elements:", sum(diag(surface\_matrix2))), file\_conn)  
  
# Суммы строк матрицы  
writeLines(paste("row sums:", rowSums(surface\_matrix2)), file\_conn)  
  
# Суммы столбцов матрицы  
writeLines(paste("col sums:", colSums(surface\_matrix2)), file\_conn)  
  
close(file\_conn)  
  
# Задание 3 ---------------------------------------------------------------  
  
# Считываем данные из файла "inputs.txt"  
params <- scan("inputs.txt", what = double())  
params

## [1] 0.5 -2.0 2.0 0.3 -1.0 1.0

# Извлекаем параметры сетки  
step\_x3 <- params[1]  
dekart\_begin\_x3 <- params[2]  
dekart\_end\_x3 <- params[3]  
step\_y3 <- params[4]  
dekart\_begin\_y3 <- params[5]  
dekart\_end\_y3 <- params[6]  
  
# Создаем координатную сетку  
x3 <- seq(from = dekart\_begin\_x3, to = dekart\_end\_x3, by = step\_x3)  
y3 <- seq(from = dekart\_begin\_y3, to = dekart\_end\_y3, by = step\_y3)  
  
# Создаем матрицу функции на сетке  
surface\_matrix3 <- outer(X = x3,  
 Y = y3,  
 FUN = function(x3,y3) Re(exp(-1i \* 0.5 \* x3 \* y3)))  
dimnames(surface\_matrix3) <- list(x3, y3)  
  
# Количество элементов матрицы  
write(paste("number of matrix elements:", length(surface\_matrix3)), "summary3.txt")  
  
file\_conn <- file("summary3.txt", "a")  
  
# Размерность строк и столбцов  
writeLines(paste("number of rows:", nrow(surface\_matrix3)), file\_conn)  
writeLines(paste("number of cols:", ncol(surface\_matrix3)), file\_conn)  
  
# Сумма элементов главной диагонали  
writeLines(paste("sum of main diag elements:", sum(diag(surface\_matrix3))), file\_conn)  
  
# Суммы строк матрицы  
writeLines(paste("row sums:", rowSums(surface\_matrix3)), file\_conn)  
  
# Суммы столбцов матрицы  
writeLines(paste("col sums:", colSums(surface\_matrix3)), file\_conn)  
  
close(file\_conn)  
  
# Машины ------------------------------------------------------------------  
  
cars\_matrix <- as.matrix(cars)  
cars\_matrix

## speed dist  
## [1,] 4 2  
## [2,] 4 10  
## [3,] 7 4  
## [4,] 7 22  
## [5,] 8 16  
## [6,] 9 10  
## [7,] 10 18  
## [8,] 10 26  
## [9,] 10 34  
## [10,] 11 17  
## [11,] 11 28  
## [12,] 12 14  
## [13,] 12 20  
## [14,] 12 24  
## [15,] 12 28  
## [16,] 13 26  
## [17,] 13 34  
## [18,] 13 34  
## [19,] 13 46  
## [20,] 14 26  
## [21,] 14 36  
## [22,] 14 60  
## [23,] 14 80  
## [24,] 15 20  
## [25,] 15 26  
## [26,] 15 54  
## [27,] 16 32  
## [28,] 16 40  
## [29,] 17 32  
## [30,] 17 40  
## [31,] 17 50  
## [32,] 18 42  
## [33,] 18 56  
## [34,] 18 76  
## [35,] 18 84  
## [36,] 19 36  
## [37,] 19 46  
## [38,] 19 68  
## [39,] 20 32  
## [40,] 20 48  
## [41,] 20 52  
## [42,] 20 56  
## [43,] 20 64  
## [44,] 22 66  
## [45,] 23 54  
## [46,] 24 70  
## [47,] 24 92  
## [48,] 24 93  
## [49,] 24 120  
## [50,] 25 85

# Задание 1 ---------------------------------------------------------------  
  
cars\_speed <- cbind(rep(1, nrow(cars\_matrix)), cars\_matrix[, 1])  
cars\_speed

## [,1] [,2]  
## [1,] 1 4  
## [2,] 1 4  
## [3,] 1 7  
## [4,] 1 7  
## [5,] 1 8  
## [6,] 1 9  
## [7,] 1 10  
## [8,] 1 10  
## [9,] 1 10  
## [10,] 1 11  
## [11,] 1 11  
## [12,] 1 12  
## [13,] 1 12  
## [14,] 1 12  
## [15,] 1 12  
## [16,] 1 13  
## [17,] 1 13  
## [18,] 1 13  
## [19,] 1 13  
## [20,] 1 14  
## [21,] 1 14  
## [22,] 1 14  
## [23,] 1 14  
## [24,] 1 15  
## [25,] 1 15  
## [26,] 1 15  
## [27,] 1 16  
## [28,] 1 16  
## [29,] 1 17  
## [30,] 1 17  
## [31,] 1 17  
## [32,] 1 18  
## [33,] 1 18  
## [34,] 1 18  
## [35,] 1 18  
## [36,] 1 19  
## [37,] 1 19  
## [38,] 1 19  
## [39,] 1 20  
## [40,] 1 20  
## [41,] 1 20  
## [42,] 1 20  
## [43,] 1 20  
## [44,] 1 22  
## [45,] 1 23  
## [46,] 1 24  
## [47,] 1 24  
## [48,] 1 24  
## [49,] 1 24  
## [50,] 1 25

cars\_dist <- cars\_matrix[, 2]  
cars\_dist

## [1] 2 10 4 22 16 10 18 26 34 17 28 14 20 24 28 26 34 34 46  
## [20] 26 36 60 80 20 26 54 32 40 32 40 50 42 56 76 84 36 46 68  
## [39] 32 48 52 56 64 66 54 70 92 93 120 85

alpha <- solve(t(cars\_speed) %\*% cars\_speed) %\*% t(cars\_speed) %\*% cars\_dist  
alpha

## [,1]  
## [1,] -17.579095  
## [2,] 3.932409

class(alpha)

## [1] "matrix" "array"

typeof(alpha)

## [1] "double"

alpha <- as.vector(alpha)  
alpha

## [1] -17.579095 3.932409

alpha\_c <- alpha[1]  
alpha\_x <- alpha[2]  
cat("alpha\_c = ", alpha\_c, "\n")

## alpha\_c = -17.57909

cat("alpha\_x = ", alpha\_x, "\n")

## alpha\_x = 3.932409

cars\_speed\_lm <- cars\_matrix[,1]  
cars\_speed\_lm

## [1] 4 4 7 7 8 9 10 10 10 11 11 12 12 12 12 13 13 13 13 14 14 14 14 15 15  
## [26] 15 16 16 17 17 17 18 18 18 18 19 19 19 20 20 20 20 20 22 23 24 24 24 24 25

cars\_dist\_lm <- alpha\_c + cars\_speed\_lm \* alpha\_x  
cars\_dist\_lm

## [1] -1.849460 -1.849460 9.947766 9.947766 13.880175 17.812584 21.744993  
## [8] 21.744993 21.744993 25.677401 25.677401 29.609810 29.609810 29.609810  
## [15] 29.609810 33.542219 33.542219 33.542219 33.542219 37.474628 37.474628  
## [22] 37.474628 37.474628 41.407036 41.407036 41.407036 45.339445 45.339445  
## [29] 49.271854 49.271854 49.271854 53.204263 53.204263 53.204263 53.204263  
## [36] 57.136672 57.136672 57.136672 61.069080 61.069080 61.069080 61.069080  
## [43] 61.069080 68.933898 72.866307 76.798715 76.798715 76.798715 76.798715  
## [50] 80.731124

dist\_residuals <- cars\_dist\_lm - cars\_dist  
dist\_residuals

## [1] -3.849460 -11.849460 5.947766 -12.052234 -2.119825 7.812584  
## [7] 3.744993 -4.255007 -12.255007 8.677401 -2.322599 15.609810  
## [13] 9.609810 5.609810 1.609810 7.542219 -0.457781 -0.457781  
## [19] -12.457781 11.474628 1.474628 -22.525372 -42.525372 21.407036  
## [25] 15.407036 -12.592964 13.339445 5.339445 17.271854 9.271854  
## [31] -0.728146 11.204263 -2.795737 -22.795737 -30.795737 21.136672  
## [37] 11.136672 -10.863328 29.069080 13.069080 9.069080 5.069080  
## [43] -2.930920 2.933898 18.866307 6.798715 -15.201285 -16.201285  
## [49] -43.201285 -4.268876

mean\_res <- mean(dist\_residuals)  
mean\_res

## [1] -1.069367e-13

sd\_res <- sd(dist\_residuals)  
sd\_res

## [1] 15.22184

sorted\_cars\_dist\_lm <- sort(cars\_dist\_lm)  
print(sorted\_cars\_dist\_lm)

## [1] -1.849460 -1.849460 9.947766 9.947766 13.880175 17.812584 21.744993  
## [8] 21.744993 21.744993 25.677401 25.677401 29.609810 29.609810 29.609810  
## [15] 29.609810 33.542219 33.542219 33.542219 33.542219 37.474628 37.474628  
## [22] 37.474628 37.474628 41.407036 41.407036 41.407036 45.339445 45.339445  
## [29] 49.271854 49.271854 49.271854 53.204263 53.204263 53.204263 53.204263  
## [36] 57.136672 57.136672 57.136672 61.069080 61.069080 61.069080 61.069080  
## [43] 61.069080 68.933898 72.866307 76.798715 76.798715 76.798715 76.798715  
## [50] 80.731124

cat("Среднее отклонение вектора dist\_residuals: ", mean\_res, "\n")

## Среднее отклонение вектора dist\_residuals: -1.069367e-13

cat("Стандартное отклонение вектора dist\_residuals: ", sd\_res, "\n")

## Стандартное отклонение вектора dist\_residuals: 15.22184