TP_0_Analyse_données

EL_Hadrami

16/10/2020

Exercice 1

```r
# order c'est pour les indices
# Creations de trois vecteurs
x <- c(1, 3, 5, 7, 9)
y <- c(2, 3, 5, 7, 11, 13)
z <- c(9, 3, 2, 5, 9, 2, 3, 9,1)
x + 2
```

```
## [1]  3  5  7  9 11
```

```r
y * 3
```

```
## [1]  6  9 15 21 33 39
```

```r
length(x)
```

```
## [1] 5
```

```r
x + y  # erreur
```

```
## Warning in x + y: la taille d'un objet plus long n'est pas multiple de la taille
## d'un objet plus court
```

```
## [1]  3  6 10 14 20 14
```

```r
sum(x > 5)
```

```
## [1] 2
```

```r
sum(x[x > 5])
```

```
## [1] 16
```

```r
sum(x > 5 | x < 3)
```

```
## [1] 3
```

```r
y[3]
```

```
## [1] 5
```

```r
y[-3]
```

```
## [1]  2  3  7 11 13
```

```r
y[x] ; (y > 7) ; y[y > 7] ; sort(z) ; sort(z, dec = TRUE) ; rev(z) ;
```

```
## [1]  2  5 11 NA NA
```

```
## [1] FALSE FALSE FALSE FALSE  TRUE  TRUE
```

```
## [1] 11 13
```

```
## [1] 1 2 2 3 3 5 9 9 9
```

```
## [1] 9 9 9 5 3 3 2 2 1
```

```
## [1] 1 9 3 2 9 5 2 3 9
```

```
order(z) ; unique(z) ; duplicated(z) ; table(z) ; rep(z, 3)
```

```
## [1] 9 3 6 2 7 4 1 5 8
```

```
## [1] 9 3 2 5 1
```

```
## [1] FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE
```

```
## z
## 1 2 3 5 9
## 1 2 2 1 3
```

```
##  [1] 9 3 2 5 9 2 3 9 1 9 3 2 5 9 2 3 9 1 9 3 2 5 9 2 3 9 1
```

Exercice 2

```
diag1 <- diag(1,nrow=9,ncol=9) # creation d'une matrice diagonale
x1 <- rep(x=1,times=9)
matrix1 <- matrix(x1,nrow=9,ncol=9)
matdiag0 <- matrix1 - diag1
```

Exercice 3

```
# Creation de deux vecteurs
vect1 <- c(1:10)
vect2 <- c(11:20)
vectconcat <- c(1,vect1[2],vect2,vect1[c(3:10)])
```

Exercice 4

```
x <- c (4.12, 1.84, 4.28, 4.23, 1.74, 2.06, 3.37, 3.83, 5.15, 3.76, 3.23, 4.87,
5.96, 2.29, 4.58)
x_extract1 <- x[(4:15)] # extraction
x_extract2 <- x[(2:14)]
x_extract3 <- x[x > 2.57 & x < 3.48]
x_extract4 <- x[x > 4.07  || x < 1.48]
indice_min <- which.min(x)
```

```
row1A <- c(-2,1,-3,-2)
row2B <- c(1,2,1,-1)
row3C <- c(-2,1,1,-1)
row4D <- c(-1,-3,1,1)
A <- matrix(c(row1A,row2B,row3C,row4D),nrow=4,ncol=4,byrow = TRUE)
rowA <- c(2,-1,3,-4)
rowB <- c(2,-2,4,-5)
rowC <- c(-2,1,3,-1)
rowD <- c(-1,-3,1,-1)
B <- matrix(c(rowA,rowB,rowC,rowD),ncol = 4,nrow = 4,byrow = T)
# Montrons que  A et B sont inversible
det(A)
```

```
## [1] 25
```

```r
det(B)
```

```
## [1] 4
```

```r
# det(A) et det(B) ne sont pas nuls donc A et B sont inversible
# l'inverse des matrices
solve(A)
```

```
##       [,1]  [,2]  [,3] [,4]
## [1,]  0.08  0.48 -0.44  0.2
## [2,] -0.24 -0.44  0.32 -0.6
## [3,] -0.12  0.28  0.16  0.2
## [4,] -0.52 -1.12  0.36 -0.8
```

```r
solve(B)
```

```
##        [,1]  [,2]  [,3] [,4]
## [1,]  -7.50  6.50 -0.50 -2.0
## [2,]   3.25 -2.75  0.25  0.5
## [3,] -10.25  8.75 -0.25 -2.5
## [4,] -12.50 10.50 -0.50 -3.0
```

```r
#2.
det(t(A))
```

```
## [1] 25
```

```r
det(solve(A))
```

```
## [1] 0.04
```

```r
1 / det(A)
```

```
## [1] 0.04
```

```r
det(A %*% B)
```

```
## [1] 100
```

```r
det(A) * det(B)
```

```
## [1] 100
```

```r
#3.
t(solve(A))
```

```
##       [,1]  [,2]  [,3]  [,4]
## [1,]  0.08 -0.24 -0.12 -0.52
## [2,]  0.48 -0.44  0.28 -1.12
## [3,] -0.44  0.32  0.16  0.36
## [4,]  0.20 -0.60  0.20 -0.80
```

```r
solve(t(A))
```

```
##       [,1]  [,2]  [,3]  [,4]
## [1,]  0.08 -0.24 -0.12 -0.52
## [2,]  0.48 -0.44  0.28 -1.12
## [3,] -0.44  0.32  0.16  0.36
## [4,]  0.20 -0.60  0.20 -0.80
```

```r
t(A %*% B)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    6    5   -3  -11
## [2,]    3   -1    4    5
## [3,]  -13   13    0  -11
## [4,]    8  -14    3   17
```

```r
t(B) %*% t(A)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    6    5   -3  -11
## [2,]    3   -1    4    5
## [3,]  -13   13    0  -11
## [4,]    8  -14    3   17
```

```r
solve(A %*% B)
```

```
##       [,1]  [,2]  [,3]  [,4]
## [1,] -1.06 -4.36  4.58 -3.90
## [2,]  0.63  2.28 -2.09  1.95
## [3,] -1.59 -6.04  6.37 -5.35
## [4,] -1.90 -7.40  7.70 -6.50
```

```r
solve(B) %*% solve(A)
```

```
##       [,1]  [,2]  [,3]  [,4]
## [1,] -1.06 -4.36  4.58 -3.90
## [2,]  0.63  2.28 -2.09  1.95
## [3,] -1.59 -6.04  6.37 -5.35
## [4,] -1.90 -7.40  7.70 -6.50
```

Exercice 6

```r
r1 <- c(1,1,3)
r2 <- c(5,2,6)
r3 <- c(-2,-1,-3)
A <- matrix(c(r1,r2,r3),nrow = 3,ncol = 3,byrow = TRUE)
# 1.
A %*% A %*% A
```

```
##      [,1] [,2] [,3]
## [1,]    0    0    0
## [2,]    0    0    0
## [3,]    0    0    0
```

```r
# Donc A est nilpotent avev n = 3
# 2 .
A[3,] <- A[1,] + A[2,]
A
```

```
##      [,1] [,2] [,3]
## [1,]    1    1    3
## [2,]    5    2    6
## [3,]    6    3    9
```

Exercice 7

```r
# 1. Creation de la  matrice X
X <- matrix(c(0.5^2, 0.5, 1, 1^2, 1, 1, 1.5^2, 1.5, 1), ncol = 3, byrow = TRUE)
det(X)
```

```
## [1] -0.25
```

```r
# donc X est inversible
# inverse de X
invX <- solve(X)
invX
```

```
##      [,1] [,2] [,3]
## [1,]    2   -4    2
## [2,]   -5    8   -3
## [3,]    3   -3    1
```

```r
# determinons a,b et c
r <- c(1,4,5)
sol <- invX %*% r
rownames(sol) <- c("a","b","c")
sol
```

```
##   [,1]
## a   -4
## b   12
## c   -4
```

```r
# 4 determination des valeurs propres
eigen(X)$values # valeurs propres
```

```
## [1]  3.10873548 -0.94393086  0.08519538
```

```r
eigen(X)$vectors # vecteurs propres
```

```
##            [,1]        [,2]       [,3]
## [1,] -0.3608737 -0.61917748  0.4165621
## [2,] -0.5337897 -0.08316023 -0.8387339
## [3,] -0.7647475  0.78083521  0.3507156
```

<u>Exercice 8</u>

```r
A <- matrix(0, nrow = 5, ncol = 5)
B  <- abs(col(A) - row(A)) + 1
B
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    2    1    2    3    4
## [3,]    3    2    1    2    3
## [4,]    4    3    2    1    2
## [5,]    5    4    3    2    1
```

```r
det(B)
```

```
## [1] 48
```

```r
# Donc B est inversible
# inverse de B
solve(B)
```

```
##               [,1]          [,2]          [,3]          [,4]          [,5]
## [1,] -4.166667e-01  5.000000e-01 -8.881784e-17  3.256654e-16  8.333333e-02
## [2,]  5.000000e-01 -1.000000e+00  5.000000e-01 -7.401487e-16  1.850372e-16
## [3,] -1.110223e-16  5.000000e-01 -1.000000e+00  5.000000e-01  0.000000e+00
## [4,]  2.775558e-17 -1.110223e-16  5.000000e-01 -1.000000e+00  5.000000e-01
```

```
## [5,]  8.333333e-02  2.465190e-32 -1.110223e-16  5.000000e-01 -4.166667e-01
```

```
# Resolution du systeme
r <- c(1,rep(2,2),3,2)
sol <- solve(B) %*% r
rownames(sol) <- c("a","b","c","d","e")
sol
```

```
##    [,1]
## a  0.75
## b -0.50
## c  0.50
## d -1.00
## e  0.75
```

Exercice 9

```
x <- 1:6
y <- 5:10
xplusy <- x + y
xplusy[xplusy > 11]
```

```
## [1] 12 14 16
```

```
# produit scalaire de x et y
prodxy <- x * y
M <- matrix(1:36, nrow = 6)
M %*% x # Calcul de M*x
```

```
##      [,1]
## [1,]  441
## [2,]  462
## [3,]  483
## [4,]  504
## [5,]  525
## [6,]  546
```

```
x %*% M # Calcul de x*M
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]   91  217  343  469  595  721
```

```
M %*% t(M) # Calcul de M*t(M)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,] 2166 2262 2358 2454 2550 2646
## [2,] 2262 2364 2466 2568 2670 2772
## [3,] 2358 2466 2574 2682 2790 2898
## [4,] 2454 2568 2682 2796 2910 3024
## [5,] 2550 2670 2790 2910 3030 3150
## [6,] 2646 2772 2898 3024 3150 3276
```

Exercice 10

```
# Creation d'une de 16 lignes et trois colonnes
r <- rep(seq(3,1,by = -1),16)
A <- matrix(r,nrow = 16,ncol = 3,byrow = TRUE)
dim(A)
```

```
## [1] 16  3
```

```r
head(A,5)
```

```
##      [,1] [,2] [,3]
## [1,]    3    2    1
## [2,]    3    2    1
## [3,]    3    2    1
## [4,]    3    2    1
## [5,]    3    2    1
```

Exercice 11

```r
Ligne1 <- c(95,68,85,72,55,86,115)
Ligne2 <- c(189,169,179,167,171,178,179)
mat <- matrix(c(Ligne1,Ligne2),nrow = 2,ncol=7,byrow = TRUE)
rownames(mat) <- c("Poids","Taille")
colnames(mat) <- c("John","Lilly","Stef","Bob","Anna", "Marik","Boris")
mat
```

```
##        John Lilly Stef Bob Anna Marik Boris
## Poids    95    68   85  72   55    86   115
## Taille  189   169  179 167  171   178   179
```

Exercice 12

```r
r1A <- c(3,1,sqrt(6))
r2A <- c(1,3,-sqrt(6))
r3A <- c(-sqrt(6),sqrt(6),2)
A <- matrix(c(r1A,r2A,r3A),nrow = 3,ncol = 3,byrow = TRUE)
A <- 1 / 4 * A
r1B <- c(-2,-1,2)
r2B <- c(2,-2,1)
r3B <- c(1,2,2)
B <- matrix(c(r1B,r2B,r3B),nrow = 3,ncol = 3,byrow = TRUE)
B <- 1 / 3 * B
A %*% t(A)
```

```
##              [,1]         [,2] [,3]
## [1,] 1.000000e+00 5.551115e-17    0
## [2,] 5.551115e-17 1.000000e+00    0
## [3,] 0.000000e+00 0.000000e+00    1
```

```r
# donc A est orthogonal
#2.verifions que inv(A)=t(A)
solve(A)
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.7500000  0.2500000 -0.6123724
## [2,] 0.2500000  0.7500000  0.6123724
## [3,] 0.6123724 -0.6123724  0.5000000
```

```r
t(A)
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.7500000  0.2500000 -0.6123724
## [2,] 0.2500000  0.7500000  0.6123724
## [3,] 0.6123724 -0.6123724  0.5000000
```

```r
# 3. Montrons que B est orthogonal
B %*% t(B)
```

```
##      [,1] [,2] [,3]
## [1,]    1    0    0
## [2,]    0    1    0
## [3,]    0    0    1
```

```r
# 4. Est ce que  que A et B commutent
ab <- A %*% B
ba <- B %*% A
ab - ba
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.8623724  0.0000000  0.8623724
## [2,] 0.0000000 -0.8623724 -0.9747449
## [3,] 0.9747449 -0.8623724  0.0000000
```

```r
# Donc A et B ne commute pas
# 5. Le determinant de A
det(A)
```

```
## [1] 1
```

```r
# 6. Valeurs et vecteurs propres de A
vpA <- eigen(A)$values
vpA
```

```
## [1] 0.5+0.8660254i 0.5-0.8660254i 1.0+0.0000000i
```

```r
vectpA <- eigen(A)$vectors
vectpA
```

```
##                 [,1]            [,2]             [,3]
## [1,]  0.0000000+0.5i  0.0000000-0.5i  7.071068e-01+0i
## [2,]  0.0000000-0.5i  0.0000000+0.5i  7.071068e-01+0i
## [3,] -0.7071068+0.0i -0.7071068+0.0i -3.716104e-18+0i
```

```r
vpB <- eigen(B)$values
vpB
```

```
## [1]  1.0000000+0.0000000i -0.8333333+0.5527708i -0.8333333-0.5527708i
```

```r
vectpB <- eigen(B)$vectors
vectpB
```

```
##              [,1]                   [,2]                   [,3]
## [1,] 0.3015113+0i  0.6741999+0.0000000i  0.6741999+0.0000000i
## [2,] 0.3015113+0i -0.0674200-0.6708204i -0.0674200+0.6708204i
## [3,] 0.9045340+0i -0.2022600+0.2236068i -0.2022600-0.2236068i
```

```r
# 7. Creation d'une nouvelle matrice
C <- A
C[3,] <- C[1,] + C[2,]
C
```

```
##      [,1] [,2]       [,3]
## [1,] 0.75 0.25  0.6123724
## [2,] 0.25 0.75 -0.6123724
## [3,] 1.00 1.00  0.0000000
```

```r
det(C)
```

```
## [1] 0
```

```
# Donc C n'est pas inversible
```