

TP R 1 - Bases du Langage et Statistiques Descriptives

L. Ziad

EST-Essaouira

Partie 1 : Prise en main de R

Exercice 1 : Opérations de base

```
1 # Calculs de base
2 (5 + 3) * 2
3 sqrt(25)
4 log(100)
5 exp(1)
6
7 # Creation de vecteurs
8 x <- c(1, 3, 5, 7, 9)
9 y <- 1:10
10 z <- seq(0, 1, by = 0.1)
11
12 # Operations sur vecteurs
13 x + 2
14 x * y[1:5]
15 sum(x)
16 mean(x)
17
```

Exercice 2 : Structures de données

```
1 # Matrices
2 mat <- matrix(1:12, nrow = 3, ncol = 4)
3 mat
4 t(mat)
5
6 # Data frames
7 df <- data.frame(
8   nom = c("Alice", "Bob", "Charlie"),
9   age = c(25, 30, 35),
10  salaire = c(3000, 4000, 5000)
11 )
12 df
13
14 # Listes
15 ma_liste <- list(
16   vecteur = 1:5,
17   matrice = mat,
18   dataframe = df
19 )
20
```

Partie 2 : Statistiques descriptives

Exercice 3 : Analyse univariee

```
1 set.seed(123)
2 notes <- rnorm(100, mean = 12, sd = 2)
3 ages <- sample(18:65, 100, replace = TRUE)
4
5 # Statistiques descriptives
6 summary(notes)
7 mean(notes)
8 sd(notes)
9 var(notes)
10 quantile(notes, probs = c(0.25, 0.5, 0.75))
11
12 # Visualisation (sans accents)
13 hist(notes, main = "Distribution des notes")
14 boxplot(notes, main = "Boite a moustaches des notes")
15
```

Exercice 4 : Analyse bivariee

```
1 set.seed(456)
2 heures_etude <- rnorm(100, 15, 5)
3 notes_finales <- 8 + 0.5 * heures_etude + rnorm(100, 0, 1)
4
5 # Correlation
6 cor(heures_etude, notes_finales)
7 cov(heures_etude, notes_finales)
8
9 # Regression lineaire
10 modele <- lm(notes_finales ~ heures_etude)
11 summary(modele)
12
13 # Visualisation
14 plot(heures_etude, notes_finales, main = "Heures etude vs Notes")
15 abline(modele, col = "red")
16
```

Partie 3 : Manipulation de donnees

Exercice 5 : Fonctions et boucles

```
1 # Fonction de statistiques
2 calcul_statistiques <- function(x) {
3   return(c(
4     moyenne = mean(x),
5     ecart_type = sd(x),
6     mediane = median(x),
7     taille = length(x)
8   ))
9 }
10
11 # Test fonction
12 donnees_test <- rnorm(50, 10, 2)
```

```
13 calcul_statistiques(donnees_test)
14
15 # apply et boucles
16 matrice_grande <- matrix(rnorm(1000), nrow = 100)
17
18 apply(matrice_grande, 2, mean)
19
20 resultats <- numeric(10)
21 for(i in 1:10) {
22   resultats[i] <- mean(matrice_grande[, i])
23 }
24
```