**Statistique en logiciel R**

**TP2 Statistique descriptive univarié**

**Exercice 1**

On a mesuré la taille (en cm) de 40 élèves d’une classe. Les résultats sont les suivants :

138 ; 164 ; 150 ; 132 ; 144 ; 125 ; 149 ; 157 ; 146 ; 158 ; 140 ; 147 ; 136 ; 148 ; 152 ; 144 ; 168 ; 126 ; 138 ; 176 ; 163 ; 119 ; 154 ; 165 ; 146 ; 173 ; 142 ; 147 ; 135 ; 153 ; 140 ; 135 ; 161 ; 145 ; 135 ; 142 ; 150 ; 156 ; 145 ; 128 ;

a) Calculer la moyenne des tailles.

> elev <- c(138, 164, 150, 132, 144, 125, 149, 157, 146, 158, 140, 147, 136, 148, 152, 144, 168, 126, 138, 176, 163, 119, 154, 165, 146, 173, 142, 147, 135, 153, 140, 135, 161, 145, 135, 142, 150, 156, 145, 128)

> mean(elev)

[1] 146.8

b) Regrouper les données en 5 classes et en 10 classes puis représenter graphiquement (par un histogramme) les résultats dans les deux cas.

en 5 classes :

> hist(elev, breaks = seq(from = min(elev), to = max(elev), length = 6), col = grey(0.8), labels = TRUE, las = 1, xlab = "Taille (cm)", ylab = "Nombre d'élèves", main = "Avec un découpage en 5 classes") -> avec5

en 10 classes :

> hist(elev, breaks = seq(from = min(elev), to = max(elev), length = 11), col = grey(0.8), labels = TRUE, las = 1, xlab = "Taille (cm)", ylab = "Nombre d'élèves", main = "Avec un découpage en 10 classes") -> avec10

c) Calculer la moyenne dans les deux cas. Pour cela nous allons utiliser la fonction **weighted.mean()** qui permet de calculer la moyenne pondérée

**Exemple d’utilisation de la fonction weighted.mean()**

> x <- c(3.7,3.3,3.5,2.8)

> wt <- c(5, 5, 4, 1)/15

> xm <- weighted.mean(x, wt) # moyenne des valeurs de x

> xm

[1] 3.453333

**Réponse de la question**

> moy5 <- weighted.mean(avec5$mids, avec5$counts)

> moy5

[1] 146.93

> moy10 <- weighted.mean(avec10$mids, avec10$counts)

> moy10

[1] 146.645

d) Tracer la boite à moustaches

boxplot(elev, horizontal=T, xlab = "Taille (cm)", main = "Taille de 40 élèves")

rug(elev,0.1)

**Exercice 2**

Dans cet exercice nous allons utiliser la fonction ave() pour calculer la moyenne conditionnée

Considérons la base de données suivante :

> data = data.frame(cbind(c(rep("homme",4),rep("femme",5)),c(1:4,3:7)))

> names(data) = c("genre","score")

> data$score = as.numeric(data$score)

> data

genre score

1 homme 1

2 homme 2

3 homme 3

4 homme 4

5 femme 3

6 femme 4

7 femme 5

8 femme 6

9 femme 7

Nous voulons calculer le score moyenne des hommes d'une part et des femmes d'autre part.

> attach(data)

> ave(score,genre)

[1] 2.5 2.5 2.5 2.5 5.0 5.0 5.0 5.0 5.0

> cbind(data,MoyCondi =ave(score,genre))

genre score MoyCondi

1 homme 1 2.5

2 homme 2 2.5

3 homme 3 2.5

4 homme 4 2.5

5 femme 3 5.0

6 femme 4 5.0

7 femme 5 5.0

8 femme 6 5.0

9 femme 7 5.0

La fonction aggregate() calcul également la moyenne conditionnée. Pour tester cette fonction nous allons utiliser la base de données data enregistrée dans le fichier « test.txt »

On utilisons la fonctions read.table(), lire et stocker les données dans une dataframe nommée data.

> data = read.table(file.choose(), header=T)

> data

Name Month Taux1 Taux2

1 Aira 1 12 23

2 Aira 2 18 73

3 Aira 3 19 45

4 Ben 1 53 19

5 Ben 2 22 87

6 Ben 3 19 45

7 Cat 1 22 87

8 Cat 2 67 43

9 Cat 3 45 32

>

Utilizer la function aggregate() pour calculer la moyenne conditionnée de Taux1 et Taux 2 selon la variable Name

> aggregate(data[, 3:4], list(data$Name), mean)

Group.1 Taux1 Taux2

1 Aira 16.33333 47.00000

2 Ben 31.33333 50.33333

3 Cat 44.66667 54.00000

>

Une autre syntaxe

> aggregate( . ~ Name, data[-2], mean)

Name Taux1 Taux2

1 Aira 16.33333 47.00000

2 Ben 31.33333 50.33333

3 Cat 44.66667 54.00000

Vous pouvez la fonction aggregate() pour calculer la variance conditionnée, l’écart-type conditionnés…, en remplaçant la fonction mean par la fonction adéquate : var, sd, sum…

Aggregate pour calculer la somme pour chaque classe

> aggregate(. ~ Name, data[-2], sum)

Name Taux1 Taux2

1 Aira 49 141

2 Ben 94 151

3 Cat 134 162

Aggregate pour calculer la variance conditionnée

> aggregate(. ~ Name, data[-2], var)

Name Taux1 Taux2

1 Aira 14.33333 628.000

2 Ben 354.33333 1177.333

3 Cat 506.33333 847.000

Aggregate pour calculer l’écart-type conditionné

> aggregate(. ~ Name, data[-2], sd)

Name Taux1 Taux2

1 Aira 3.785939 25.05993

2 Ben 18.823744 34.31229

3 Cat 22.501852 29.10326

Aggregate pour calculer l’effectif cumulé de chaque classe

> aggregate(. ~ Name, data[-2], cumsum)

Name Taux1.1 Taux1.2 Taux1.3 Taux2.1 Taux2.2 Taux2.3

1 Aira 12 30 49 23 96 141

2 Ben 53 75 94 19 106 151

3 Cat 22 89 134 87 130 162

Vous pouvez faire la même chose par la fonction by

> by(data[,"Taux1"], data[,"Name"], mean)

data[, "Name"]: Aira

[1] 16.33333

------------------------------------------------------------

data[, "Name"]: Ben

[1] 31.33333

------------------------------------------------------------

data[, "Name"]: Cat

[1] 44.66667

>

**Autres Fonctions :**

prop.table() #donne directemenet les fréquences

round() # pour arrondir les valeurs

par () # comme subplot de matlab