T. D. nº 1
Statistiques descriptives sur une série de mesures univariée

1 Quelques commandes sous R qui peuvent vous permettre de faire de la description empirique

1.1 Statistique d'ordre

```
>x = rnorm(10) (Echantillon i.i.d.)
>y = sort(x) (Statistique d'ordre)
```

1.2 Fonction de répartition empirique

```
>x=rnorm(100)
>n=length(x)
>plot(sort(x),1:n/n,type="s",ylim=c(0,1),xlab="",ylab="")
>?pnorm
>curve(pnorm(x,0,1),add=T,col="blue")
```

1.3 Histogramme

```
>x=rnorm(100)
>hist(x,breaks=20)
>hist(x,breaks=20,freq=F,col="cyan")
>curve(dnorm(x),add=T,col="darkblue")
>x=rnorm(50)
>h=hist(x, plot=F)
>h$breaks
>h$counts
>hist
```

1.4 Boxplot

```
>x=rnorm(100)
>par(mfcol=c(2,2),bg="lightcyan")
>boxplot(x)
>boxplot(x,horizontal=T)
>boxplot(x,col="red")
>boxplot(x,col="orange",border="darkblue",lwd=2)
```

1.5 Boxplots en parallèle

```
>x=rnorm(100)
>y=(rnorm(400))^2-1
>z=rnorm(50)^3
>par(bg="lightcyan")
>boxplot(x,y,z,col=c("blue","white","red"),
+border=c("black","darkblue"),lwd=1.5)
```

1.6 QQ-plots

```
>x = norm(100)\\
>y =(rnorm(400))^2-1
>z=rnorm(200,m=4,sd=5)
>par(bg="lightcyan",mfrow=c(2,2))
>qqplot(x,y,pch=21,bg="red",fg="darkblue",lwd=2)
>qqplot(x,z,pch=21,bg="red",fg="darkblue",lwd=2)
>qqnorm(y,pch=21,bg="orange",fg="darkblue",lwd=2)
>qqline(y,pch=21,col="blue",lwd=2)
>qqnorm(z,pch=21,bg="orange",fg="darkblue",lwd=2)
>qqline(z,pch=21,col="blue",lwd=2)
```

2 Exercices

Exercice 1 Jeu de données Temps de travail pour un emploi à plein temps. Traiter avec le logiciel `R`, l'exemple sur le nombre d'heures travaillées par semaine en Europe (2006), à savoir calculer les statistiques descriptives annoncées dans le cours et tracer la boîte à moustaches. Quelle(s) différence(s) faîtes-vous avec les résumés numériques et la représentation graphique donnés dans les transparents du cours ?

Exercice 2 Jeu de données Mesures, pages 90 à 95 du livre « Initiation à la statistique avec `R` ».

a) Télécharger le package `BioStatR` en cliquant en haut sur la barre d'outils sur `Packages`, puis choisir `Installer le(s) package(s)`. Puis choisissez le pays le plus proche de l'endroit où vous vous trouvez. Dans notre cas, vous avez cinq choix : France (IGH Montpellier), France (Lyon 1), France (IBCP Lyon), France (Paris 1), France (IRSN Paris) puis placer le curseur sur `BioStatR` et enfin cliquer sur `Ok`.

b) Taper la ligne de commande :
`>library(BioStatR)`

c) Pour afficher le jeu de données `Mesures`, taper la ligne de commande :
`>Mesures`

d) Pour afficher les premières lignes de ce fichier, taper la ligne de commande :
`>head(Mesures)`
Remarque : Par défaut, la fonction `head()` affiche les six premières lignes. Si vous souhaitez afficher les 10 premières lignes, il faut taper la ligne de commande suivante :
`>head(Mesures,10)`
Pour afficher les dernières lignes de ce fichier, taper la ligne de commande :
`>tail(Mesures)`
Remarque : Par défaut, la fonction `tail()` affiche les six dernières lignes. Si vous souhaitez afficher les 10 dernières lignes, il faut taper la ligne de commande suivante :
`>tail(Mesures,10)`

e) Si vous tapez la commande suivante :
`>str(Mesures)`
vous voyez apparaître le mot `Factor`, qui représente la classe de la variable `espece`. En effet, si vous tapez la ligne de commande suivante :
`>class(Mesures$espece)`
`R` vous renvoie :
`[1] "factor"`
Savez-vous ce que représente ce terme ? En fait, il indique que la variable `espece` est une variable qualitative.
`R` vous renseigne aussi sur son nombre de modalités (« levels » en anglais) : la variable `espece` en a quatre. Comment obtenez-vous les noms des quatre espèces ? Une première idée serait d'utiliser la fonction `names` et donc de taper la ligne de commande suivante :
`>names(Mesures$espece)`
mais R renvoie
`NULL`
Savez-vous pourquoi ? En fait, la fonction `names` renvoie le nom des colonnes du jeu de données `Mesures`. En effet, tapez la ligne de commande suivante :
`>names(Mesures)`
et `R` renvoie
`[1] "masse" "taille" "espece"` Donc, si vous voulez le nom des modalités de la variable espece, il faut utiliser la fonction `levels` et donc taper la ligne de commande suivante : `>levels(Mesures$espece)`
`[1] "bignone" "glycine blanche" "glycine violette" "laurier rose"`
Quelques mois après, le jardinier a complété son premier fichier `Mesures` en un fichier `Mesures5` avec deux nouvelles variables qui sont :
— la masse sèche, relevée sur les 252 « haricots »,
— et le nombre de graines contenues dans les gousses des glycines blanches et violettes.
Donc, regardez de quoi est constitué le fichier `Mesures5` en tapant la ligne de commande suivante :
`>str(Mesures5)`

Exercice 3 Fonction factor, pages 143 à 145 du livre « Initiation à la statistique avec R ».
Dans cet exercice, vous allez découvrir comment fonctionne la fonction `factor` *que vous devez connaître pour la suite.*

Sur trois variétés de pommes notées 1, 2 et 3, la jutosité de chaque pomme est relevée. La jutosité est un indice compris entre 0 et 10. Il y a quatre pommes par variété qui ont été testées. La variété 1 est la Golden Delicious, la variété 2 est la pomme Calville et la variété 3 est la Belle de Boskoop. Vraisemblablement, la question que vous pourriez vous poser serait : « quelle est la variété de pomme la plus juteuse ? » Vous ne chercherez pas à répondre à cette question ici. En effet, il s'agit d'une application d'une technique statistique connue sous le nom d'analyse de la variance que vous ne connaissez pas encore. Le but de cet exercice est de vous montrer comment vous servir de la fonction `factor`. Les résultats obtenus sont inscrits dans le tableau suivant :

Variété de pomme	Jutosité	Variété de pomme	Jutosité
1	4	2	7
1	6	2	6
1	3	3	8
1	5	3	6
2	7	3	5
2	8	3	6

a) Rentrez les données sous `R` en introduisant deux variables :
— une première variable que vous noterez `Variete`
— et une seconde variable que vous noterez `Jutosite`.
À l'issue de cette opération, construisez un `data.frame` dont le nom est `Pommes`.

b) Donnez la structure du jeu de données `Pommes` que vous venez de construire à la question précédente. Que constatez-vous ?
Il faut donc transformer la variable `Variete` en un `factor`.

c) Pour transformer un vecteur de type numérique ou entier, vous pouvez utiliser la fonction `factor`. Ainsi pour transformer la variable `Variete` qui est pour l'instant de mode `numeric`, vous tapez la ligne de commande suivante :
```
> Variete<-factor(Variete)
```
puis
```
> Pommes<-data.frame(Variete,Jutosite)
> rm(Variete)
> rm(Jutosite)
```
Quelle est la nature du jeu de données `Pommes` ? Quels sont les modes des deux variables qui constituent le jeu de données `Pommes` ?
Remarque : `rm` pour « remove ».

d) Vous auriez pu procéder autrement. Cette seconde façon est beaucoup plus rapide et vous êtes invité à vous en servir dès que vous savez qu'une variable

dans votre jeu de données est un facteur.
Tapez les lignes de commandes suivantes :

```
> Variete<-factor(c(rep(1,4),rep(2,4),rep(3,4)))
> Jutosite<-c(4,6,3,5,7,8,7,6,8,6,5,6)
> Pommes<-data.frame(Variete,Jutosite)
```

Qu'obtenez-vous ? Avez-vous le même résultat qu'auparavant, c'est-à-dire la même structure pour le jeu de données `Pommes` ?

e) Il vous est conseillé, au moins dans les premiers temps de votre apprentissage de la statistique, de ne pas utiliser des nombres pour les niveaux de votre facteur, mais plutôt des lettres. Pour cela, vous utiliserez l'option `labels` dans la fonction `factor`.
Vous allez donner un label aux valeurs numériques `1`, `2` et `3`, à savoir `1` devient `V1`, `2` devient `V2` et `3` devient `V3`, `V` pour `Variete`. Pour cela, tapez les lignes de commande suivantes :

```
> Variete<-factor(c(rep(1,4),rep(2,4),rep(3,4)),
+labels=c("V1","V2","V3"))
> Jutosite<-c(4,6,3,5,7,8,7,6,8,6,5,6)
> Pommes<-data.frame(Variete,Jutosite)
```

Qu'obtenez-vous ? Il y a quelque chose qui a changé. Pouvez-vous dire quoi ?

f) Enfin, il existe une fonction `as.factor` qui permet d'arriver au même résultat.
Tapez les lignes de commande suivantes :

```
> Variete<-as.factor(c(rep(1,4),rep(2,4),rep(3,4)))
> Jutosite<-c(4,6,3,5,7,8,7,6,8,6,5,6)
> Pommes<-data.frame(Variete,Jutosite)
```

Vérifiez bien que vous obtenez le même résultat qui est attendu.

g) Calculez les moyennes pour chacun des groupes défini par la variable `Variete` en utilisant la fonction `tapply` :

```
> tapply(Jutosite,Variete,mean)
```

Procédez de même pour obtenir l'écart-type, les quantiles ou appliquer la fonction `summary` à chacun des groupes défini par le facteur `Variete`.

Exercice 4 Comment grouper des données ?, pages 145 et 146 du livre « Initiation à la statistique avec R ».
Vous allez vous intéresser ici à la variable masse du jeu de données `Mesures`*. Dans les rappels de cours, vous avez vu comment grouper ces données automatiquement à l'aide de l'une des trois règles dues à Sturges, Scott et Freedman-Diaconis. Vous allez voir comment grouper des données suivant différents critères.*

a) Groupez les données en 5 classes à l'aide de l'option `breaks=5` de la fonction `hist`. Que se passe-t-il si vous cherchez à en obtenir seulement 4 ?

b) Groupez les données en utilisant les classes suivantes $[0;5]$, $]5;10]$, $]10;15]$, $]15;20]$ et $]20;50]$ à l'aide de l'option `breaks=c(0,5,10,15,20,50)` de la fonction `hist`.

c) Comparez le résultat obtenu avec :
```
> brk <- c(0,5,10,15,20,50)
> table(cut(Mesures$masse, brk))
> data.frame(table(cut(Mesures$masse, brk)))
```

d) Si vous cherchez à créer des groupes dont les effectifs sont équilibrés, vous pouvez par exemple utiliser la fonction `cut2` de la bibliothèque `Hmisc`. Après avoir téléchargé et installé cette bibliothèque, commentez les lignes de code suivantes et en particulier le rôle des options `g` et `m`.
```
> library(Hmisc)
> brk <- c(0,5,10,15,20,50)
> res <- cut2(Mesures$masse, brk)
> table(res)
> table(cut2(Mesures$masse, g=10))
> table(cut2(Mesures$masse, m=50))
```

Exercice 5 Jeu de données Europe, page 147 du livre « Initiation à la statistique avec R ».
Le but de cet exercice est de calculer des résumés numériques et de tracer une boîte à moustaches.

a) Affichez les six premières lignes de ce jeu de données qui est disponible dans la bibliothèque `BioStatR`.

b) De quoi est constitué ce jeu de données ? C'est-à-dire : combien de variables composent ce jeu de données ? quelle est la nature de ces variables ? Combien d'unités statistiques sont présentes dans ce jeu de données ?

c) Quelle est la classe et la taille de ce jeu de données ?

d) Donnez la moyenne, la valeur minimale, la valeur maximale, la médiane et la ou les classes modale(s) de la variable `Duree`.

e) Donnez l'écart-type corrigé, le coefficient de variation et l'étendue de la variable `Duree` .

f) Tracez la boîte à moustaches de la variable `Duree` en mettant un label pour l'axe vertical qui est « Durée en heures ».
Représentez sur cette même boîte la moyenne.

g) Sauvegardez la boîte à moustaches au format .pdf et au format .ps en utilisant les fonctions `pdf` et `postscript`.

Exercice 6 Données brutes ou groupement en classes, page 150 du livre « Initiation à la statistique avec R ».
Lorsque vous étudiez une série statistique sur un caractère quantitatif qui comporte un grand nombre de valeurs, il est suggéré de la grouper par classes puis ensuite remplacer chaque classe par son milieu. Mais les résultats en sont légèrement modifiés,

ce que vous pouvez imaginer. D'ailleurs certains auteurs suggèrent des corrections pour certaines des caractéristiques.

Exemple : *En ce qui concerne la variance certains auteurs et en particulier Couty, Debord et Fredon dans leur livre « Mini manuel de probabilités et statistique », aux éditions Dunod, 2007, suggèrent la correction de Sheppard.*

Le but de cet exercice est d'illustrer un groupement en classes défini préalablement par l'utilisateur.

Le jeu de données ci-dessous est extrait de Couty, Debord et Fredon, « Mini manuel de probabilités et statistique », aux éditions Dunod, 2007.

Considérez une série statistique de 60 taux d'hémoglobine dans le sang (g/L) mesurés chez des adultes présumés en bonne santé. La série est rangée par valeurs croissantes et l'ordre dans lequel les données ont été observées n'a pas été conservé.

Femmes	105	110	112	112	118	119	120	120	125	126
	127	128	130	132	133	134	135	138	138	138
	138	142	145	148	148	150	151	154	154	158
Hommes	141	144	146	148	149	150	150	151	153	153
	153	154	155	156	156	160	160	160	163	164
	164	165	166	168	168	170	172	172	176	179

a) Créez deux vecteurs : un vecteur `Femmes` et un vecteur `Hommes` qui contiennent chacun les données brutes.

b) Considérez le groupement en classes suivant :

$$[104;114],]114;124],]124;134],]134;144],]144;154],]154;164],$$
$$]164;174],]174;184].$$

Pour chacune des deux séries : femmes et hommes, déterminez les effectifs et les fréquences de chaque classe.

c) Effectuez une représentation graphique adaptée des deux distributions groupées en classe de la question 1.

d) Calculez les moyennes des trois distributions initiales : ensemble, femmes, hommes.

e) Calculez les moyennes des trois distributions (ensemble, femmes, hommes) après le groupement en classes de la question 1., en remplaçant chaque classe par son milieu.

f) Calculez les médianes des trois distributions initiales : ensemble, femmes, hommes.

g) Calculez l'écart interquartile pour chacune des trois distributions initiales : ensemble, femmes, hommes.

h) Calculez les variances corrigées et les écarts-types corrigés des trois distributions initiales : ensemble, femmes, hommes.

i) Calculez les variances et les écarts-types des trois distributions après le regroupement en classes de la question 1., en remplaçant chaque classe par son milieu.

j) Pour la distribution des femmes, calculez les caractéristiques de forme de Fisher.