TD2: Variables qualitatives

.

Mars 2023

Introduction : Ce T.D. a pour but de vous faire assimiler les représentations numériques et graphiques des statistiques descriptives univariée pour les variables qualitatives.

Objectif de ce T.D. :
- Manipuler les données.
- Faire des résumés numériques et/ou graphiques.

Consignes pour ce T.D. :
- Suivre pas à pas les étapes et voir ce qui se passe.
- Ne pas hésiter à utiliser l'aide en ligne de `R`.
- Vous ne comprendrez peut-être pas tous les détails mais la meilleure chose à faire est de taper le code et de voir le résultat produit. Soyez curieux et n'hésitez pas à le modifier pour voir " ce qu'il se passe ".

Quelques remarques :
- Le symbole `#` signifie le début d'un commentaire.
- Lorsque vous travaillez sous `R`, il peut être intéressant de conserver les résultats et les graphiques de vos analyses. Le plus simple, dans un premier temps, est de les enregistrer dans un document word à l'aide du copier / coller. Pour ce faire, aller dans le menu " Plots ", sélectionner" Copy image ". Noter que les graphes peuvent être réduits ou agrandis sans déformation.
- Parfois le signe `+` peut apparaitre en début de ligne de commande de `R`. Ne le tapez pas svp. Il est là pour rappeler qu'une ligne a été coupée et que nous sommes en début de ligne.

Exercice 1 Fichier de données : iris. (*Adapté des notes de cours de Myriam et Frédéric Bertrand*)

Le logiciel `R` est un ensemble de bibliothèques de fonctions appelées "packages". Chaque bibliothèque contient des jeux de données. Pour connaître par exemple les jeux de données contenus dans le package `datasets`, écrire l'instruction suivante :

```
data(package = "datasets")
```

Le résultat apparaît dans une fenêtre `R` data sets. En voici un extrait : `Data sets in package 'datasets':`

```
AirPassengers                  Monthly Airline Passenger Numbers
1949-1960
BJsales                        Sales Data with Leading Indicator
BJsales.lead (BJsales)         Sales Data with Leading Indicator
```

```
BOD                             Biochemical Oxygen Demand
...
iris                            Edgar Anderson's Iris Data
```

1. Noter la présence du fichier `iris` dans la liste ci-dessus. Les données de cefichier sont célèbres. Elles ont été collectées par Edgar Anderson. Le fichier donne les mesures en centimètres des variables suivantes :

- longueur du sépale (Sepal.Length),
- largeur du sépale (Sepal.Width),
- longueur du pétale (Petal.Length),
- largeur du pétale (Petal.Width)
  pour trois espèces d'iris qui sont les :

1. Iris setosa,
2. Iris versicolor et
3. Iris virginica.

Sir R.A. Fisher 2 a utilisé ces données pour construire des combinaisons linéaires des variables permettant de séparer au mieux les trois espèces d'iris.

2. Pour analyser le fichier iris, il faut le charger. Quelle est l'instruction qu'il faut taper pour charger ce fichier ?

3. Taper une à une chacune des instructions ci-dessous et noter le résultat obtenu si possible.
   Attention : le logiciel `R` n'est pas indifférent aux majuscules et aux minuscules, comme nous l'avons déjà souligné dans le T.D. 1.

```
iris
dim(iris)
names(iris)
```

Quelle(s) différence(s) faites-vous avec la commande ?

```
str(iris)
```

4. Taper les lignes de commande suivantes :

```
iris$Petal.Length
iris$Species
```

Qu'observez-vous ?

5. La dernière colonne du fichier `iris` contient le nom des espèces réparties en trois catégories : setosa, versicolor et virginica. Pour accéder à celle-ci, il faut utiliser l'instruction `iris$Species`, comme vous venez de le constater à la question précédente. Nous disons alors que la dernière colonne contient une variable qualitative à trois modalités ou à trois niveaux appelés "levels" par le logiciel `R`. La fonction `levels` appliquée à la colonne `iris$Species` donne les modalités de la variable. En effet, il suffit de taper :

```
levels(iris$Species)
```

Pour résumer l'information contenue dans cette variable, vous utiliserez l'instruction suivante :

```
summary(iris$Species)
```

Quel est le résultat que vous obtenez ?

6. Cette dernière information peut être obtenue en construisant un tableau (`table`) comptabilisant le nombre d'individus par modalité. Pour ce faire, taper l'instruction suivante :

```
table(iris$Species)
```

Comparer avec le résultat obtenu à la question précédente.

7. R permet également de réaliser des résumés graphiques. Lorsqu'une instruction graphique est lancée, une nouvelle fenêtre dans le menu "plot", s'ouvre. Les représentations graphiques liées aux variables qualitatives sont : — le diagramme circulaire ou encore le camembert, voire la fonction (`pie()`)
   — la diagramme en bâtons, voire la fonction (`barplot()`).
   Taper les lignes de commande suivantes :

```
pie(table(iris$Species))
barplot(table(iris$Species))
```

8. Il existe une fonction, la fonction `par()`, permettant de découper la fenêtre graphique de deux façons : `par(mfrow=c(nl,nc))` ou `par(mfcol=c(nl,nc))`, où `nl` définit le nombre de graphiques en lignes, `nc` définit le nombre de graphiques en colonnes, `mfrow` signifie que l'ordre d'entrée des graphiques s'effectue selon les lignes et `mfcol` signifie que l'ordre d'entrée des graphiques s'effectue selon les colonnes.

Supposons que vous vouliez représenter six graphiques dans une seule fenêtre en deux lignes et trois colonnes. La première instruction conduit à entrer les graphiques selon l'ordre :

1	2	3
4	5	6

La seconde instruction conduit à entrer les graphiques selon l'ordre :

1	3	5
2	4	6

Deux botanistes se sont également intéressés aux iris et ont collecté les espèces suivantes :

```
collection1<-rep(c("setosa","versicolor","virginica"), c(15,19,12))
collection2<-rep(c("setosa","versicolor","virginica"), c(22,27,17))
```

En utilisant la fonction `par(mfrow=c(2,2))`,
1. Construire les camemberts de ces deux nouvelles distributions. Commenter.
2. Construire les diagrammes en bâtons de ces deux nouvelles distributions. Com-

menter.
3. Discuter des avantages et des inconvénients de ces deux types de représentations.

Exercice2: Variations génétiques dans les populations humaines. (*Notes de cours de Christophe Ambroise*)
1. Charger le jeu de données `hdpg` du package `ade4` et lire son descriptif.
2. Nous considérerons le tableau `hdpg$ind` qui décrit l'échantillon des `1066` individus de l'étude.
3. Combien de populations différentes participent à l'étude ?
4. Dresser les tableaux des effectifs des variable population, région et sexe.
5. Transformer ces tableaux en tableaux de fréquences.
6. Représenter vos tableaux d'effectifs par des diagrammes en bâton, et par des camemberts.
7. Commenter les représentations.

Exercice 3: Fonction factor (*Adapté des notes de cours de Myriam et Frédéric Bertrand*)

Dans cet exercice, vous allez découvrir comment fonctionne la fonction `factor`.

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

1. 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.

2. 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`.

3. 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".

4. 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 commande 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` ?

5. 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 ?

6. 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.

7. 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`.