Fiche TD avec le logiciel R : tpRg4

BIO2049L Statistique pour la Biologie et Bioinformatique
Exploration graphique d'un jeu de données

I. Amat, D. Fouchet, L. Keurinck, V. Lacroix, J.R. Lobry, A. Mary, L. Nicvert, M.C. Venner & S. Venner

L'objectif de cette séance est de vous faire utiliser le logiciel R pour procéder à l'exploration préliminaire d'un jeu de données. Le but sera essentiellement d'apprendre à utiliser quelques commandes utiles pour extraire et visualiser des données.

Contents

1 Lancement de R et calculs élémentaires

1.1 Lancement de R en salle de TP

Il existe plusieurs interfaces permettant de jouer avec le logiciel R. Nous vous conseillons d'utiliser `RStudio` qui est disponible sous `Ubuntu` en salle de TP. Le logiciel R fonctionne comme une calculatrice à laquelle on donne des ordres. Par exemple, la commande `print(pi)` donne l'ordre d'afficher la valeur approximative de π. Dans ce document toutes les commandes R sont données en rouge, il est inutile de perdre votre temps à les recopier : vous pouvez simplement les copier/coller à partir du PDF dont l'URL est donnée en pied-de-page de ce document. Votre premier exercice consiste donc à ouvrir ce PDF puis à copier/coller la commande ci-après dans votre console (fenêtre située en bas à gauche dans `RStudio`), taper un retour chariot pour l'exécuter et vérifier que vous obtenez le bon résultat, en bleu dans ce document.

```
print(pi)
```
```
[1] 3.141593
```

Une fonction dans R est toujours du type `nomfonction(argument1, argument2, ...)`. Selon la fonction, il y aura un nombre variable d'arguments (=paramètres) à renseigner, dont certains seront obligatoires et d'autres optionnels.

Si vous n'êtes pas arrivés à reproduire ce résultat, faites vous aider par votre chargé de TD ou par un collègue de votre groupe de TD. Pour la suite nous supposerons que cette étape est acquise.

1.2 Calculs élémentaires

Vous pouvez utiliser R comme une calculatrice. La syntaxe des opérateurs arithmétiques usuels se trouve facilement en consultant la documentation du logiciel. Voici quelques exemples :

```
3 + 5
```
```
[1] 8
```
```
3*5
```
```
[1] 15
```
```
9/3
```
```
[1] 3
```
```
3^2
```
```
[1] 9
```

On peut utiliser l'opérateur d'affectation « `<-` », composé des deux caractères « `<` » et « `-` », pour à la fois créer un objet et y ranger une (ou plusieurs) valeurs. Par exemple, pour mettre la valeur 6 dans l'objet de nom `w` :

```
w <- 6
w + w
```

```
[1] 12
```

Donnez le code R permettant de calculer 6 fois la valeur de `w`, vous devez obtenir le résultat suivant :

```
[1] 36
```

Réponse 1 :

L'opérateur deux points « : » permet de générer des séries d'entiers consécutifs :

```
1:12
 [1]  1  2  3  4  5  6  7  8  9 10 11 12
```

Donnez le code R permettant de générer la série suivante :

```
 [1] -5 -4 -3 -2 -1  0  1  2  3  4  5
```

Réponse 2:

La fonction `c()` permet de construire « à la main » n'importe quel vecteur, par exemple :

```
x <- c(2, 5, 8)
x
[1] 2 5 8
```

Taper `?c` dans la console pour découvrir à quoi correspond la fonction `c()`

Donnez le code R permettant de ranger les valeurs 7, 8 et 9 dans le vecteur y :

```
[1] 7 8 9
```

Réponse 3:

Il y a en général plusieurs façons de coder pour un même résultat

Les opérations arithmétiques usuelles fonctionnent directement avec des vecteurs, elles se font élément par élément :

```
x + y
[1]  9 13 17
```

L'addition de deux vecteurs numériques de même longueur (ici, 3) produit un vecteur numérique de même longueur contenant les sommes deux à deux des valeurs de rang identique

2 Importation de données dans R

2.1 Les données sur les notes des étudiants

Vous avez sans doute l'habitude de manipuler vos données avec des outils de type tableur. Nous allons vous montrer comment importer facilement ce type de données sous R à l'aide d'un exemple détaillé. Puis ce sera à vous de jouer avec un autre jeu de données. Dans votre butineur de toile favori, copiez/collez l'adresse suivante[1] :

`https://goo.gl/bU3nJK`

Il s'agit d'un tableau de données où, par convention universellement suivie en statistiques, les individus (ici des étudiants) sont disposés en ligne et les variables en colonne. Pour pouvoir importer ces données dans R, nous allons utiliser un fichier de données brut au format « `tsv` ». Ce fichier est un simple fichier texte dans lequel chaque ligne correspond à une ligne du tableau et chaque case est séparée par une tabulation. Ce fichier est disponible à l'adresse ci-dessous :

`https://pbil.univ-lyon1.fr/R/donnees/bsbi/mathsv.tsv`

Vous pouvez également obtenir ce fichier à partir du tableur en téléchargeant une copie de ces données « `tsv` ». Cependant nous n'allons pas avoir besoin de télécharger ces données brutes puisque R est capable d'importer des données à partir d'une adresse distante (l'url du fichier qui vous intéresse). Pour importer un fichier de données, qu'il soit présent en local sur votre disque dur ou qu'il soit disponible via une adresse distante, on utilise la fonction « `read.table` ». Cette fonction permet de créer objet R de type « `DataFrame` » à partir d'un fichier de données. Une `DataFrame` peut être vu comme un tableau similaire à celui que vous avez visualisé dans le tableur. Dans le logiciel `RStudio` tapez la commande suivante:

```
etudiants <- read.table("https://pbil.univ-lyon1.fr/R/donnees/bsbi/mathsv.tsv",
  header = TRUE, dec = ',')
```

> La commande précédente demande à R de créer un objet de type `DataFrame` à partir du fichier « `mathsv.tsv` » et de l'enregistrer dans une variable nommée « `etudiants` ». Vous pouvez bien sûr choisir un nom de variable différent. En plus de l'adresse du fichier, nous renseignons également deux arguments supplémentaires à la fonction.
>
> **`header = TRUE`** permet de préciser que la première ligne du fichier correspond aux noms des colonnes et ne doit pas être considérée comme une ligne de donnée normale.
>
> **`dec = ','`** permet de préciser que les valeurs numériques décimales utilisent la notation française, c'est à dire que nous utiliserons la virgule (et non pas le point anglo-saxon) pour séparer la partie entière de la partie décimale d'un nombre.

[1] C'est une version compactée de `https://docs.google.com/spreadsheets/d/18O0QZrseB0F6PTpCFZ0tH2rw7xIK5PU8loJJR3evY_U/edit#gid=324731599`

etudiants

Le jeu de données `etudiants` est un échantillon de 220 étudiants inscrits à l'université Claude Bernard - Lyon 1 ayant eu un baccalauréat de la série S et ayant été notés à la première session du contrôle terminal de l'unité d'enseignement « MathSV » au semestre de printemps 2017. Les étudiants sont caractérisés par les 12 variables suivantes :

1° `note`, une variable quantitative donnant la note de l'étudiant en MathSV sur une échelle croissante allant, des moins bons résultats aux meilleurs, de 0 à 20 ;

2° `sequence`, une variable qualitative nominale indiquant si l'étudiant est inscrit en séquence `2` ou `3` ;

3° `ffsu`, une variable qualitative nominale à deux modalité `OUI` ou `NON` indiquant si l'étudiant est inscrit en FFSU[a] de niveau 2 ;

4° `gEPS`, une variable qualitative ordonnée à deux modalités indiquant le goût autodéclaré de l'étudiant pour le sport (`-` ou `+`) ;

5° `sexe`, une variable qualitative nominale donnant le sexe de l'étudiant (`M` ou `F`) ;

6° `abs`, une variable qualitative ordonnée à deux modalités donnant le niveau d'absentéisme de l'étudiant lors des séances de TD de MathSV (`0-1` ou `>1`) ;

7° `gMaths`, une variable qualitative ordonnée à trois modalités indiquant le goût autodéclaré de l'étudiant pour les mathématiques (`-`, `0` ou `+`) ;

8° `gPhys`, une variable qualitative ordonnée à trois modalités indiquant le goût autodéclaré de l'étudiant pour la physique (`-`, `0` ou `+`) ;

9° `mBac`, une variable qualitative ordonnée à trois modalités donnant la mention autodéclarée de l'étudiant au baccalauréat (`<12`, `12-14` ou `>14`) ;

10° `nMaths`, une variable qualitative ordonnée à quatre modalités donnant la note autodéclarée en mathématiques de l'étudiant au baccalauréat (`<10`, `10-12`, `12-14`, ou `>14`) ;

11° `nPhys`, une variable qualitative ordonnée à quatre modalités donnant la note autodéclarée en physique de l'étudiant au baccalauréat (`<10`, `10-12`, `12-14`, ou `>14`) ;

12° `nBPSVT`, une variable quantitative donnant la note de l'étudiant en « Bases de Physique pour les Sciences de la Vie et de la Terre » sur une échelle croissante allant de 0 à 20.

[a] Acronyme de la « Fédération Française du Sport Universitaire ».

2.2 Les données sur la masse des bébés

Importez dans R les données disponibles à l'adresse donnée ci-dessous :

`https://pbil.univ-lyon1.fr/R/donnees/bsbi/baby.tsv`

Vous placerez ces données dans un objet nommé `bebes`. Les données sont décrites dans l'encart page 7. Vérifiez avant de passer à la suite que vous avez bien les objets `bebes` et `etudiants` présents dans votre environnement (fenêtre en haut à droite de votre écran sous Rstudio).

bebes

Le jeu de données `bebes` est un échantillon de 1173 nouveau-nés caractérisés par les 8 variables suivantes :

1° `bwt`, la masse du bébé à la naissance exprimée en kg ;

2° `weight`, la masse de la mère, au début de la grossesse, exprimée en kg ;

3° `height`, la taille de la mère exprimée en cm[a] ;

4° `age`, l'âge de la mère exprimé en années ;

5° `gestation`, la durée de la grossesse exprimée en jours ;

6° `parity`, la parité de la mère dans son sens technique en gynécologie obstétrique : `TRUE` si c'est son premier accouchement donnant un enfant vivant, `FALSE` dans le cas contraire.

7° `smoke`, une variable indicatrice du tabagisme de la mère : `TRUE` si elle fume, `FALSE` si elle ne fume pas ;

8° `tension`, la tension artérielle moyenne de la mère au cours de la grossesse.

[a] Attention aux unités si vous voulez calculer l'indice de masse corporelle des mères : il faut diviser cette valeur par 100 pour l'avoir en m.

3 Extraction des données utiles

3.1 Extraction de colonnes (variables : après la virgule)

On peut accéder aux valeurs d'une colonne par sa position ou bien par son nom. Voici quatre façons de récupérer les données de la 5e colonne donnant le sexe des étudiants. Notez que dans le cas de l'opérateur crochet « `[ , ]` », tout ce qui a trait aux colonnes se trouve à droite de la virgule :

```
etudiants[ , 5]
```

```
  [1] "M" "F" "M" "F" "F" "F" "M" "M" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "F" "F"
 [21] "M" "M" "F" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "F" "F" "M" "F" "M" "M" "M"
 [41] "F" "F" "F" "M" "M" "F" "F" "M" "F" "F" "M" "M" "F" "M" "M" "F" "M" "F" "M" "M"
 [61] "F" "F" "M" "F" "F" "F" "F" "M" "F" "F" "M" "M" "M" "F" "F" "F" "F" "M" "F" "F"
 [81] "F" "M" "F" "M" "F" "F" "F" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F" "M"
[101] "F" "F" "M" "M" "F" "F" "F" "F" "M" "F" "F" "F" "F" "F" "M" "M" "F" "F" "M" "F"
```

```
[121] "F" "M" "F" "M" "F" "M" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "M" "F" "F" "M"
[141] "F" "F" "F" "F" "F" "F" "F" "F" "F" "M" "F" "F" "F" "M" "F" "F" "F" "M" "M" "M"
[161] "M" "M" "F" "F" "M" "F" "F" "M" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F"
[181] "M" "F" "F" "M" "F" "M" "M" "F" "M" "F" "M" "M" "M" "F" "M" "F" "M" "F" "F" "M"
[201] "M" "F" "F" "M" "F" "M" "F" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "M" "M" "M"
```

```
etudiants[ , "sexe"]
```

```
  [1] "M" "F" "M" "F" "F" "F" "M" "M" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "F" "F"
 [21] "M" "M" "F" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "F" "F" "M" "F" "M" "M" "M"
 [41] "F" "F" "F" "M" "M" "F" "F" "M" "F" "F" "M" "M" "F" "M" "M" "F" "M" "F" "M" "M"
 [61] "F" "F" "M" "F" "F" "F" "F" "M" "F" "F" "M" "M" "M" "F" "F" "F" "F" "M" "F" "F"
 [81] "F" "M" "F" "M" "F" "F" "F" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F" "M"
[101] "F" "F" "M" "M" "F" "F" "F" "F" "M" "F" "F" "F" "F" "F" "M" "M" "F" "F" "M" "F"
[121] "F" "M" "F" "M" "F" "M" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "M" "F" "F" "M"
[141] "F" "F" "F" "F" "F" "F" "F" "F" "F" "F" "M" "F" "F" "F" "M" "F" "F" "F" "M" "M"
[161] "M" "M" "F" "F" "M" "F" "F" "M" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F"
[181] "M" "F" "F" "M" "F" "M" "M" "F" "M" "F" "M" "M" "M" "F" "M" "F" "M" "F" "F" "M"
[201] "M" "F" "F" "M" "F" "M" "F" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "M" "M" "M"
```

```
etudiants$sexe
```

```
  [1] "M" "F" "M" "F" "F" "F" "M" "M" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "F" "F"
 [21] "M" "M" "F" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "F" "F" "M" "F" "M" "M" "M"
 [41] "F" "F" "F" "M" "M" "F" "F" "M" "F" "F" "M" "M" "F" "M" "M" "F" "M" "F" "M" "M"
 [61] "F" "F" "M" "F" "F" "F" "F" "M" "F" "F" "M" "M" "M" "F" "F" "F" "F" "M" "F" "F"
 [81] "F" "M" "F" "M" "F" "F" "F" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F" "M"
[101] "F" "F" "M" "M" "F" "F" "F" "F" "M" "F" "F" "F" "F" "F" "M" "M" "F" "F" "M" "F"
[121] "F" "M" "F" "M" "F" "M" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "M" "F" "F" "M"
[141] "F" "F" "F" "F" "F" "F" "F" "F" "F" "F" "M" "F" "F" "F" "M" "F" "F" "F" "M" "M"
[161] "M" "M" "F" "F" "M" "F" "F" "M" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F"
[181] "M" "F" "F" "M" "F" "M" "M" "F" "M" "F" "M" "M" "M" "F" "M" "F" "M" "F" "F" "M"
[201] "M" "F" "F" "M" "F" "M" "F" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "M" "M" "M"
```

```
with(etudiants, sexe)
```

```
  [1] "M" "F" "M" "F" "F" "F" "M" "M" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "F" "F"
 [21] "M" "M" "F" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "F" "F" "M" "F" "M" "M" "M"
 [41] "F" "F" "F" "M" "M" "F" "F" "M" "F" "F" "M" "M" "F" "M" "M" "F" "M" "F" "M" "M"
 [61] "F" "F" "M" "F" "F" "F" "F" "M" "F" "F" "M" "M" "M" "F" "F" "F" "F" "M" "F" "F"
 [81] "F" "M" "F" "M" "F" "F" "F" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F" "M"
[101] "F" "F" "M" "M" "F" "F" "F" "F" "M" "F" "F" "F" "F" "F" "M" "M" "F" "F" "M" "F"
[121] "F" "M" "F" "M" "F" "M" "F" "F" "F" "F" "F" "F" "F" "M" "M" "F" "M" "F" "F" "M"
[141] "F" "F" "F" "F" "F" "F" "F" "F" "F" "F" "M" "F" "F" "F" "M" "F" "F" "F" "M" "M"
[161] "M" "M" "F" "F" "M" "F" "F" "M" "F" "M" "M" "F" "M" "M" "F" "F" "M" "F" "F" "F"
[181] "M" "F" "F" "M" "F" "M" "M" "F" "M" "F" "M" "M" "M" "F" "M" "F" "M" "F" "F" "M"
[201] "M" "F" "F" "M" "F" "M" "F" "M" "M" "F" "M" "F" "F" "M" "F" "M" "M" "M" "M" "M"
```

Donnez le code R pour extraire le numéro de séquence à laquelle les étudiants étaient inscrits. Vous devez obtenir le résultat suivant :

```
  [1] 2 2 3 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 3 3 2 3 3 3 3 3 2 2 2 2 2 2 2 3 2 3
 [41] 2 3 2 3 2 2 2 2 3 3 2 3 2 2 2 2 3 2 3 2 2 2 2 2 2 2 3 3 3 2 2 2 2 2 2 3 2 3 2 2
 [81] 2 2 3 2 2 2 2 2 2 2 2 2 2 3 3 2 3 3 2 2 2 2 2 3 2 2 2 2 2 2 2 3 2 2 2 3 2 2 3 2
[121] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 2 3 2 2 2 3 2 2 2 2 2 2 2 2 3 3 2
[161] 2 2 2 3 2 3 2 3 2 2 2 3 2 3 2 3 2 2 2 2 3 2 3 2 2 2 2 2 3 2 2 2 3 3 2 2 3 2 2 3 2 2
[201] 2 2 2 3 2 3 2 3 2 2 2 2 2 2 3 2 3 3 3 3 2
```

Réponse 4:

L'opérateur deux points « : » permet d'extraire une plage de colonnes consécutives, par exemple pour extraire les colonnes 5 à 7 :

```
etudiants[ , 5:7]
```

Donnez le code R pour extraire les colonnes 1 à 4 du jeu de données `etudiants` :

Réponse 5:

La fonction `c()` permet d'extraire des colonnes dans un ordre arbitraire, par exemple pour extraire les colonnes 7, 5 et 2 :

```
etudiants[ , c(7, 5, 2)]
```

Donnez le code R pour extraire les colonnes 1, 4 et 8 du jeu de données `etudiants` :

Réponse 6:

3.2 Extraction de lignes (individus : avant la virgule)

On cherche en général à extraire les individus qui sastisfont certaines propriétés. Notez que dans le cas de l'opérateur crochet « `[ , ]` », tout ce qui a trait aux lignes se trouve à gauche de la virgule. Par exemple, pour avoir le 66e étudiant :

```
etudiants[66, ]
```

```
   note sequence ffsu gEPS sexe abs gMaths gPhys mBac nMaths nPhys nBPSVT
66 16.15        2  NON    -    F 0-1      +     0  >14    >14 10-12  10.85
```

Pour extraire les étudiants qui ont eu une note strictement supérieure à 19/20 :

```
etudiants[etudiants$note > 19, ]
```

```
    note sequence ffsu gEPS sexe abs gMaths gPhys  mBac nMaths nPhys nBPSVT
40  20.00        3  NON    -    M  >1      +     + 12-14    >14   >14   8.55
47  20.00        2  NON    +    F 0-1      0     0   >14    >14   >14   9.80
61  19.62        2  NON    -    F 0-1      +     0   >14    >14   >14  12.66
67  19.62        3  NON    +    F 0-1      0     0 12-14  10-12 10-12  10.61
78  20.00        3  NON    +    M 0-1      0     0 12-14  10-12   >14   7.30
83  20.00        3  NON    -    F 0-1      +     +   >14    >14   >14  10.05
110 20.00        2  OUI    +    F  >1      +     +   >14    >14   >14  10.07
149 20.00        3  NON    +    F 0-1      +     0   >14    >14 12-14   9.01
152 20.00        2  NON    +    F 0-1      +     0   >14    >14   >14  13.60
154 20.00        2  OUI    +    F 0-1      +     0 12-14    >14   >14  13.47
156 20.00        2  NON    +    F 0-1      +     0 12-14  12-14   >14  11.28
160 20.00        2  NON    +    M 0-1      0     0   >14  12-14 12-14   9.30
163 20.00        2  NON    -    F 0-1      0     0   >14    >14   >14  15.72
169 20.00        2  OUI    +    F 0-1      +     +   >14    >14   >14  10.02
194 20.00        2  NON    -    F 0-1      0     0 12-14  12-14   >14   8.65
199 19.10        2  OUI    +    F 0-1      +     0 12-14  10-12 12-14  12.77
219 20.00        3  NON    -    M 0-1      0     + 12-14  12-14 12-14   4.31
```

Donnez le code R pour extraire les étudiants qui ont eu une note strictement inférieure à 1/20. Vous devez obtenir le résultat suivant :

```
    note sequence ffsu gEPS sexe abs gMaths gPhys  mBac nMaths nPhys nBPSVT
3   0.77        3  NON    -    M  >1      0     0   <12  12-14 10-12  10.03
25  0.77        3  NON    -    F  >1      +     0   <12  12-14   >14   1.13
28  0.00        3  NON    -    F  >1      -     0   <12    <10 12-14   1.17
44  0.00        3  NON    +    M  >1      0     + 12-14  10-12 12-14   3.61
52  0.77        3  NON    -    M  >1      0     0   <12    <10 12-14   0.88
55  0.00        2  NON    +    M  >1      0     -   <12    <10 12-14   0.26
64  0.77        2  NON    +    F  >1      -     + 12-14    <10 12-14   0.88
90  0.00        2  NON    -    M  >1      -     0   <12    <10   >14   0.65
172 0.00        2  NON    +    F  >1      0     0   <12  12-14   <10   7.01
182 0.00        3  NON    +    F 0-1      0     0   <12    <10 10-12   4.97
```

Réponse 7:

On peut combiner plusieurs conditions avec les opérateurs logiques. Par exemple, pour avoir les étudiants de sexe masculin qui ont eu une note strictement supérieure à 19/20 :

```
etudiants[etudiants$note > 19 & etudiants$sexe == "M", ]
```

```
    note sequence ffsu gEPS sexe abs gMaths gPhys  mBac nMaths nPhys nBPSVT
40    20        3  NON    -    M  >1      +     + 12-14    >14   >14   8.55
78    20        3  NON    +    M 0-1      0     0 12-14   10-12  >14   7.30
160   20        2  NON    +    M 0-1      0     0   >14   12-14 12-14   9.30
219   20        3  NON    -    M 0-1      0     + 12-14   12-14 12-14   4.31
```

Notez que la commande `with()` permet d'obtenir la même chose avec une écriture plus compacte, donc plus lisible :

```
with(etudiants, etudiants[note > 19 & sexe == "M", ])
```

```
    note sequence ffsu gEPS sexe abs gMaths gPhys  mBac nMaths nPhys nBPSVT
40    20        3  NON    -    M  >1      +     + 12-14    >14   >14   8.55
78    20        3  NON    +    M 0-1      0     0 12-14   10-12  >14   7.30
160   20        2  NON    +    M 0-1      0     0   >14   12-14 12-14   9.30
219   20        3  NON    -    M 0-1      0     + 12-14   12-14 12-14   4.31
```

Donnez le code pour extraire les étudiants de sexe féminin qui ont eu une note strictement inférieure à 1/20. Vous devez obtenir le résultat suivant :

```
    note sequence ffsu gEPS sexe abs gMaths gPhys  mBac nMaths nPhys nBPSVT
25  0.77        3  NON    -    F  >1      +     0   <12   12-14   >14   1.13
28  0.00        3  NON    -    F  >1      -     0   <12     <10 12-14   1.17
64  0.77        2  NON    +    F  >1      -     + 12-14     <10 12-14   0.88
172 0.00        2  NON    +    F  >1      0     0   <12   12-14   <10   7.01
182 0.00        3  NON    +    F 0-1      0     0   <12     <10 10-12   4.97
```

Réponse 8:

Comme dans le cas des colonnes, on peut utiliser l'opérateur deux points « : » pour extraire des lignes consécutives et la fonction `c()` pour extraire des lignes arbitraires, par exemple :

```
etudiants[1:5, ]
```

```
  note sequence ffsu gEPS sexe abs gMaths gPhys  mBac nMaths nPhys nBPSVT
1 4.23        2  NON    +    M  >1      -     +   <12   10-12   <10   8.10
2 2.31        2  NON    -    F 0-1      -     +   <12     <10   >14   6.17
3 0.77        3  NON    -    M  >1      0     0   <12   12-14 10-12  10.03
4 12.31       2  NON    -    F 0-1      0     + 12-14   12-14   <10   8.25
5 9.62        2  NON    +    F 0-1      0     0 12-14   12-14 12-14   8.01
```

```
etudiants[c(1, 10, 144), ]
```

```
    note sequence ffsu gEPS sexe abs gMaths gPhys mBac nMaths nPhys nBPSVT
1   4.23        2  NON    +    M  >1      -     +  <12   10-12   <10   8.10
10  7.69        2  NON    -    M 0-1      -     0  <12     <10 12-14  10.03
144 15.77       2  NON    +    F 0-1      +     -  <12     >14   <10  11.95
```

3.3 Extraction simultanée de lignes et de colonnes : les variables que je veux sur les individus que je veux

On peut combiner les approches précédentes pour extraire directement les données qui nous intéressent. Par exemple, on aimerait connaitre la séquence des étudiants de sexe féminin qui ont eu une note strictement supérieure à 19/20 :

```
etudiants[etudiants$note > 19 & etudiants$sexe == "F", "sequence"]
```

```
[1] 2 2 3 3 2 3 2 2 2 2 2 2 2
```

On peut extraire plusieurs colonnes d'un coup. Par exemple, il est possible d'extraire les variables `note`, et `sexe` en plus de la variable `sequence` pour vérifier que nous avons bien la séquence des étudiants de sexe féminin ayant eu plus de 19/20 :

```
etudiants[etudiants$note > 19 & etudiants$sexe == "F", c("note", "sexe", "sequence")]
```

```
    note sexe sequence
47  20.00    F        2
61  19.62    F        2
67  19.62    F        3
83  20.00    F        3
110 20.00    F        2
149 20.00    F        3
152 20.00    F        2
154 20.00    F        2
156 20.00    F        2
163 20.00    F        2
169 20.00    F        2
194 20.00    F        2
199 19.10    F        2
```

Donnez le code R pour extraire la note, le sexe et la mention au bacalauréat des étudiants de sexe masculin qui ont eu une note strictement inférieure à 1/20. Vous devez obtenir le résultat suivant :

```
   note sexe  mBac
3  0.77    M   <12
44 0.00    M 12-14
52 0.77    M   <12
55 0.00    M   <12
90 0.00    M   <12
```

Réponse 9:

4 Générer une variable et l'ajouter à un objet de type `DataFrame`

Le logiciel R manipule directement des données vectorielles, par exemple pour calculer automatiquement pour chaque étudiant la moyenne entre la note de MathSV et celle BPSVT il suffit d'écrire :

```
(etudiants$note + etudiants$nBPSVT)/2
```

```
  [1]  6.165  4.240  5.400 10.280  8.815  9.305  6.845  9.835  3.925  8.860 10.545
 [12]  9.785  9.265  9.310 12.690  7.965  7.465  4.675  4.325  4.940  5.370 10.155
 [23]  6.370  5.965  0.950  7.535  6.625  0.585  5.835  8.510  4.960  9.670  7.025
 [34]  9.600  9.185  9.465  7.245 13.610  9.205 14.275  5.150  8.935  9.365  1.805
 [45]  7.180 10.290 14.900 16.595  3.860  5.475  6.895  0.825 14.125  6.060  0.130
 [56]  5.440 13.695  6.180 15.225  7.190 16.140 15.770 11.430  0.825  3.490 13.500
 [67] 15.115  9.105 13.325 10.505  4.095  7.095  5.300  8.850  4.050  4.480 12.290
 [78] 13.650  6.650 11.865  9.235  3.945 15.025  8.185 10.375 10.000 10.810  9.830
 [89]  4.550  0.325  4.780  7.955  6.510  9.970 11.125  6.725  8.305  7.540  2.270
[100] 10.750  8.705  9.550 11.435  5.335 13.765  6.680  3.805 13.485  2.615 15.035
[111]  3.170  4.775  3.975 10.535  6.045  7.130  9.445  1.395 10.315 11.420  9.520
[122] 10.020  4.230  6.245  6.075  9.615 13.015  9.910 16.390  4.190  5.365 13.035
[133]  6.665  4.730 13.385  6.825 10.105  6.285  5.275  9.930  4.585  6.955  3.715
[144] 13.860  8.980 11.375  2.410 12.590 14.505 10.545  8.140 16.800  6.320 16.735
[155]  5.860 15.640 16.665 15.190 10.655 14.650  5.860  7.215 17.860 12.735 14.435
[166] 10.765 13.305  4.515 15.010  5.490 10.695  3.505  5.560  7.075 10.575  4.935
[177]  5.085  8.870  5.330  7.290  5.465  2.485  5.675  3.655 10.885 12.175  8.070
[188] 10.380  5.765  1.190 10.145 11.575  9.690 14.325  6.840 14.905  7.875  9.805
[199] 15.935 10.935 13.275  7.530  5.485  5.650  9.740  3.860  9.650  8.250 10.450
[210]  7.660  9.200  4.005  4.865  7.485  2.640  8.665 16.965  4.010 12.155  7.730
```

L'opération équivalente dans un tableur consisterait à entrer dans une nouvelle colonne la formule calculant la moyenne des deux notes, puis à étendre cette formule à l'ensemble des individus. Pour que l'analogie soit complète, il faudrait ajouter une nouvelle colonne dans l'objet `etudiants`, si on décide de l'appeler `moyenne`, le code R réalisant cette opération est :

```
etudiants$moyenne <-(etudiants$note + etudiants$nBPSVT)/2
```

L'indice de masse corporelle, IMC, le plus utilisé chez l'homme est celui proposé par Adolphe Quételet et défini par le rapport,

$$\text{IMC} = \frac{\text{masse}}{\text{taille}^2}$$

où la masse est exprimée en kg et la taille en m. Donnez le code R permettant de créer une nouvelle colonne `IMC` dans le jeu de données `bébés` donnant l'indice de masse corporelle des mères. On rappelle que la description des variables et des unités employées est donnée dans l'encart page 7. Pour les 50 premières mères vous devez trouver les valeurs suivantes :

```
 [1] 18.85536 23.89062 20.34766 20.18089 17.53590 33.56087 24.01704 23.56712 22.62994
[10] 18.80969 21.94141 23.37314 19.28514 28.12195 23.73998 21.44189 25.16444 26.64706
[19] 23.62902 21.49490 28.25752 24.12856 28.52941 20.93065 19.21515 22.83948 30.02320
[28] 21.44189 21.96511 20.45472 28.43967 20.46648 21.14222 20.58604 20.98998 20.99874
[37] 16.80222 17.32544 31.16582 32.32250 23.18087 22.30154 20.08768 24.95868 18.12000
[46] 24.45914 21.73243 24.51197 25.38793 21.42650
```

Réponse 10:

L'utilisation de scripts est très commode dès que l'on dépasse une ligne de texte. C'est généralement le cas quand on utilise les fonctions graphiques de R car de nombreuses options sont disponibles. Dans les cadres réservés aux réponses dans la suite du document vous pouvez recopier le code de la solution ou plus simplement donner le nom du fichier dans lequel vous avez sauvegardé votre solution.

5 Visualisation d'une variable à la fois

5.1 Cas d'une variable qualitative

5.1.1 Les diagrammes en bâtons

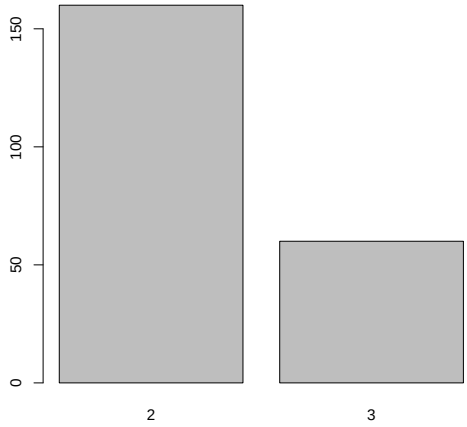
Une variable qualitative est caractérisée par les fréquences de ses modalités. La fonction `table()` permet d'effectuer cette opération, par exemple pour la séquence des étudiants :

```
table(etudiants$sequence)
```

```
  2   3
160  60
```

Une représentation en bâtons permet de bien visualiser la part relative des différentes modalités :

```
barplot(table(etudiants$sequence))
```



Ci-dessus, la fonction `barplot()` possède un unique argument, soit le nombre d'étudiants inscrits dans chacune des séquences 1 et 3: renseigner cet argument est un minimum pour exécuter une figure.
Il s'agit maintenant d'étoffer un peu la figure: essayez successivement ces différents arguments optionnels:

```
barplot(table(etudiants$sequence),
        main = "Répartition des étudiants entre les deux séquences")
```

puis

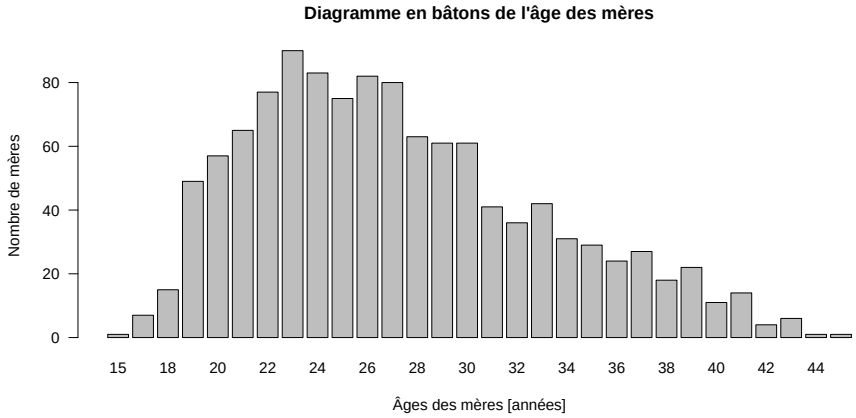
```
barplot(table(etudiants$sequence),
        xlab = "Numéro de séquence",
        ylab = "Nombre d'étudiants")
```

et encore

```
barplot(table(etudiants$sequence),
        main = "Répartition des étudiants entre les deux séquences",
        xlab = "Numéro de séquence",
        ylab = "Nombre d'étudiants",
        las = 1)
```

Renseignez-vous sur la fonction `barplot()`: il y a encore bien d'autres arguments optionnels que vous pouvez inclure, toujours en les séparant les uns des autres par des virgules dans la parenthèse.

DONNEZ le code R permettant de reproduire à l'identique de la figure ci-après le diagramme en bâtons de l'âge des mères dans le jeu de données `bébés` :



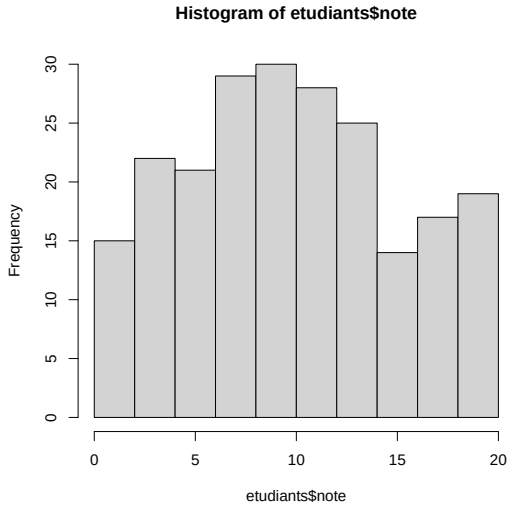
Réponse 11:

5.2 Cas d'une variable quantitative

5.2.1 Histogramme

La fonction `hist()` permet de tracer des histogrammes, par exemple pour représenter la distribution des notes en MathSV :

```
hist(etudiants$note)
```



Là encore, on a fait le minimum syndical pour exécuter la fonction `hist()` et le résultat est un peu brut: un titre par défaut pas foufou, légendes d'axes itou...

Cette fonction possède en réalité, tout comme la fonction `barplot`, de nombreuses options dont quelques unes sont mises en oeuvre ci-après. Essayez successivement les commandes ci-dessous et observez le résultat (jouez au jeu de la différence):

```
hist(etudiants$note, proba = FALSE)
```

et

```
hist(etudiants$note, proba = TRUE)
```

L'option `proba = FALSE`, qui est la valeur par défaut, fournit des hauteurs de barres d'histogramme dont la somme vaut N (soit les N notes contenues dans l'objet `etudiants$note`). Bien. L'option `proba = TRUE` permet quant à elle d'avoir une surface totale des barres d'histogramme égale à 1 (ce qui implique que l'axe des y représente une densité de probabilité et non plus un effectif ni une fréquence). Mais à quoi cela peut-il bien nous servir? Réponse dans le paragraphe qui suit...

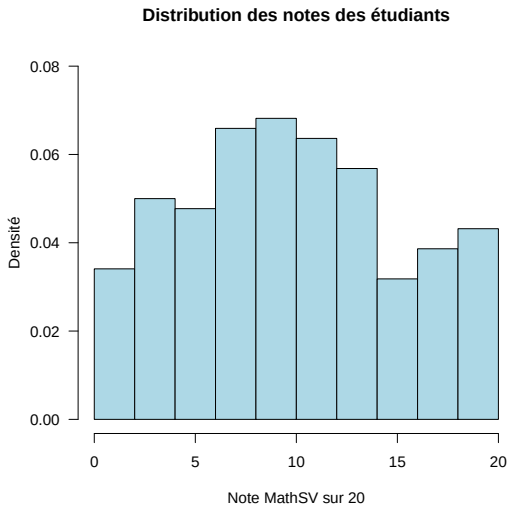
Jouez encore un peu!

```
hist(etudiants$note, proba = TRUE, breaks=13)
```

et

```
hist(etudiants$note, proba = TRUE, breaks=2)
```

```
hist(etudiants$note,
      main = "Distribution des notes des étudiants",
      xlab = "Note MathSV sur 20",
      ylab = "Densité", col = "lightblue",
      las = 1, proba = TRUE, ylim = c(0, 0.08))
```



Renseignez-vous sur la fonction `hist()` en tapant `?hist` dans la console: plusieurs arguments optionnels peuvent être inclus et personnalisés.

5.2.2 Histogramme et examen graphique de la normalité (Pour aller plus loin)

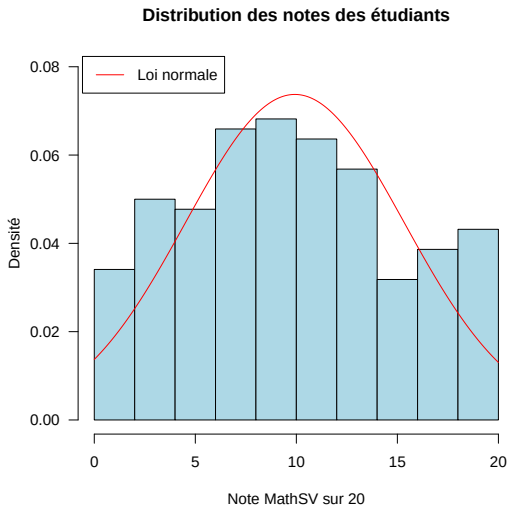
L'option `proba = TRUE` de la fonction `hist()` va nous permettre de confronter les données observées à une fonction de densité de probabilité de référence, par exemple la loi normale $\mathcal{N}(\mu, \sigma)$. On va en effet superposer une courbe de Gauss sur l'histogramme et examiner graphiquement la normalité de la distribution des notes de math! L'estimation de la moyenne de la population, $\hat{\mu}$, et l'estimation de l'écart-type de la population, $\hat{\sigma}$, sont données par les fonctions `mean()` et `sd()`, respectivement.

```
mean(etudiants$note)
```
```
[1] 9.926182
```

```
sd(etudiants$note)
```
```
[1] 5.410764
```

Si la variable `note` est normalement distribuée, la loi normale la mieux ajustée aux données sera munie de la moyenne et de l'écart-type estimés à partir de l'échantillon. On va superposer la courbe de Gauss correspondante dans l'histogramme existant.

```
hist(etudiants$note,
      main = "Distribution des notes des étudiants",
      xlab = "Note MathSV sur 20",
      ylab = "Densité", col = "lightblue",
      las = 1, proba = TRUE, ylim = c(0, 0.08))
x <- seq(from = 0, to = 20, length = 200)
lines(x, dnorm(x, mean(etudiants$note), sd(etudiants$note)), col = "red")
legend("topleft", inset = 0.01, legend = "Loi normale", lty = 1, col = "red")
```



On doit admettre que la commande est un peu complexe, avec plusieurs fonctions nouvelles et emboîtées les unes dans les autres. Décortiquons les dernières lignes en tapant les commandes ci-dessous dans la console:

1. la fonction `seq()`, `min()` et `max()`

```
?seq
x
min(x)
max(x)
seq(-1, 5, length=13)
```

2. la fonction `dnorm()`

```
?dnorm
dnorm(0)
dnorm(0, mean=0, sd=1)
dnorm(1, mean=0, sd=1)
dnorm(c(0,1),mean=0, sd=1)
dnorm(x, mean=mean(etudiants$note), sd=sd(etudiants$note))
length(dnorm(x, mean=mean(etudiants$note), sd=sd(etudiants$note)))
```

`dnorm()` calcule la densité de probabilité (=la projection sur l'axe des ordonnées) d'un quantile -ou de plusieurs- de la loi normale de moyenne `mean` et d'écart-type `sd`.

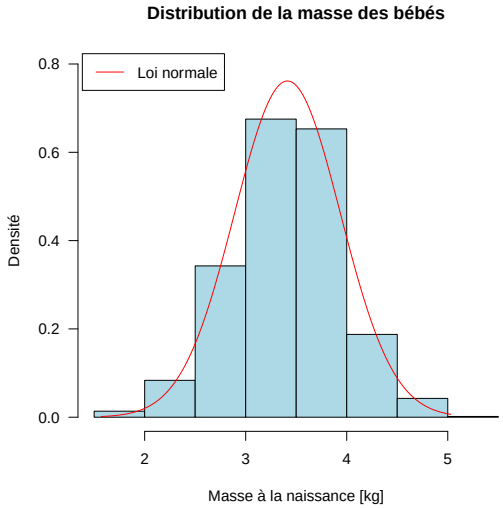
3. la fonction `lines()` et la fonction `legend()`

```
?lines
plot(0:10, -10:0, pch=20, col=grey(0.5))
lines(c(0,2), c(-10,-8), lty=1, col="red")
lines(c(2,2), c(-12,-8), lty=2, col="black")
lines(c(-1,2), c(-8,-8), lty=3, col="black")
?legend
```

`lines()` relie par un segment de droite un ou plusieurs points dont les coordonnées (x, y) sont regroupées dans les deux premiers arguments. Cette fonction n'est opérationnelle que dans un graphique existant.

`legend()` ajoute une légende dans un graphique existant.

Donnez le code R permettant de produire la représentation graphique suivante :

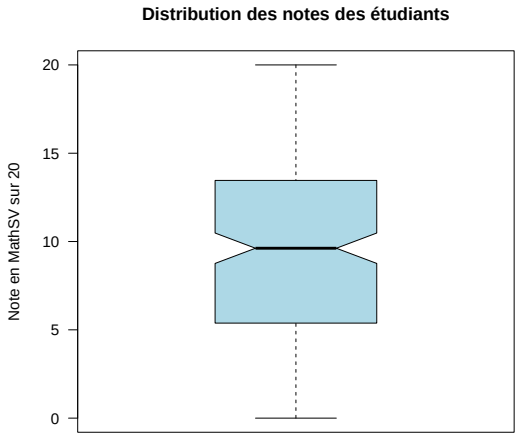


Réponse 12:

5.2.3 Boîte à moustaches

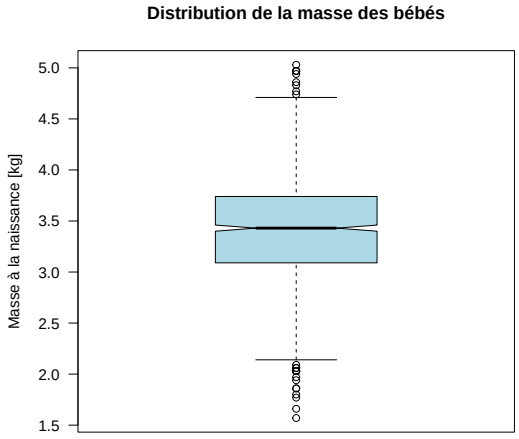
Une autre possibilité pour représenter les variables quantitatives est celle des « boîtes à moustaches ». Cette fois, c'est la fonction `boxplot()` qui va être utile pour résumer les statistiques descriptives non paramétriques d'une variable :

```
boxplot(etudiants$note,
        main = "Distribution des notes des étudiants",
        ylab = "Note en MathSV sur 20",
        las = 1, notch = TRUE,
        col = "lightblue")
```



Dans la figure qui précède, situez la médiane des notes de l'échantillon, de même que les quartiles inférieur et supérieur. L'option `notch = TRUE` va tailler des encoches dans la boîte pour situer un intervalle de confiance à 95 % de la médiane de la valeur pour la population d'origine.

Donnez le code R permettant de produire la représentation graphique suivante :



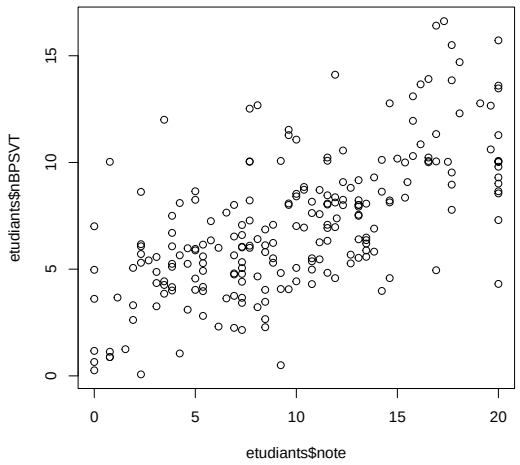
Réponse 13:

6 Visualisation de deux variables à la fois

6.1 Deux variables quantitatives

La fonction `plot(x, y)` permet de produire des nuages de points dont les coordonnées en abscisse sont données par `x` et celles en ordonnée par `y`, par exemple pour comparer la note de MathSV avec celle de BPSVT :

```
plot(etudiants$note, etudiants$nBPSVT)
```

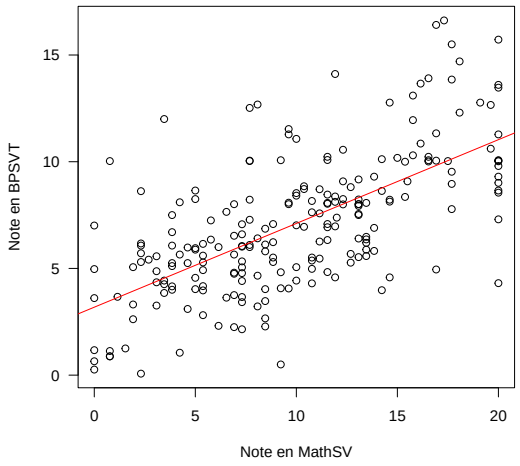


Essayez aussi cette commande:

```
plot(etudiants$nBPSVT~etudiants$note)
```

On peut enrichir ce graphique en tirant parti des options de la fonction `plot()` et en utilisant la fonction `abline()` pour ajouter la droite de régression linéaire :

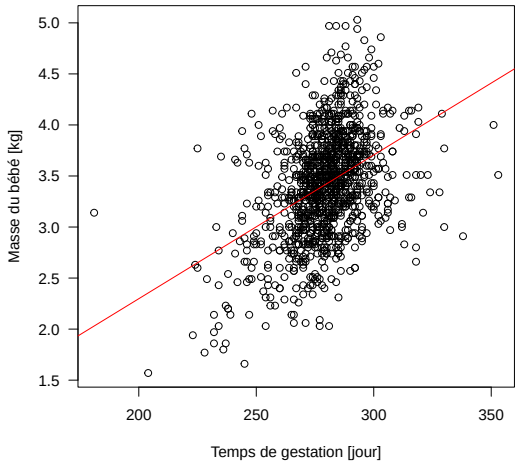
```
# Dessinons le nuage de points
#
plot(etudiants$note, etudiants$nBPSVT, las = 1,
      xlab = "Note en MathSV", ylab = "Note en BPSVT",
      main = "")
# Ajoutons la droite de régression
#
abline(lm(nBPSVT~note, etudiants), col = "red")
```



Vous verrez plus en détail dans le fascicule 2 la fonction `lm()`, laquelle apparaît ici imbriquée dans la fonction `abline()`. En attendant, n'oubliez pas de consulter `?abline`.

Le sigle # visible dans le script R ci-dessus indique que tout enchainement de caractères qui suit à sa droite **et** sur la même ligne sera interprété comme un commentaire, et ne sera pas exécuté comme du code. Cela vous permet d'annoter votre script et donc de fournir un aide-mémoire de l'utilité de l'écriture de la commande qui précède.

Donnez le code R permettant de produire la représentation graphique suivante :



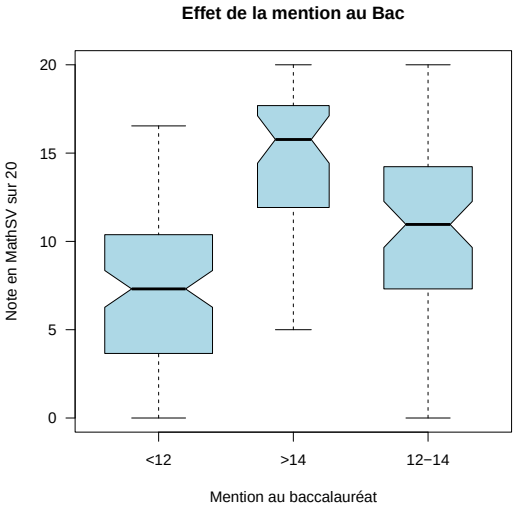
Réponse 14:

Nous reverrons plus en détail ces graphiques dans le fascicule 2 qui porte sur le traitement statistique des données.

6.2 Quantitatif-Qualitatif

La fonction `boxplot()` permet très facilement de visualiser l'effet d'une variable qualitative sur une variable quantitative. Voyons par exemple l'effet de la mention au baccalauréat sur la note en MathSV. La notation `note~mBac` se lit comme « la note en fonction de la mention au baccalauréat ». L'option `varwidth = TRUE` va imposer que la surface des boîtes soit proportionnelle aux effectifs des étudiants, ce qui permet de détecter s'il y a des déséquilibres entre les groupes.

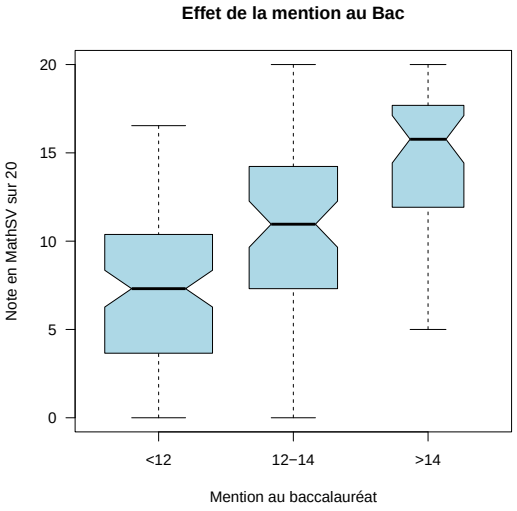
```
boxplot(note~mBac, etudiants, main = "Effet de la mention au Bac",
        xlab = "Mention au baccalauréat",
        ylab = "Note en MathSV sur 20",
        las = 1, col = "lightblue", varwidth = TRUE,
        notch = TRUE)
```



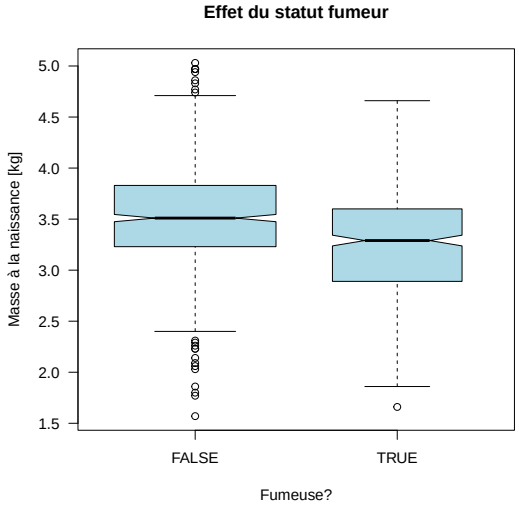
Cette représentation graphique n'est pas satisfaisante parce qu'on souhaiterait que la mention au baccalauréat, en abscisse, soit *ordonnée*. Quand on importe des données sous R tout ce qui n'est pas strictement numérique, *e.g.* comme une chaîne de caractères, est converti par défaut en des modalités d'une variable qualitative nominale, lesquelles sont représentées dans leur ordre alpha-numérique.

Pour modifier cet ordre, une manière de procéder est d'appliquer la fonction `ordered()` au facteur `mBac`:

```
etudiants$mBac <- ordered(etudiants$mBac, levels = c("<12", "12-14", ">14"))
boxplot(note~mBac, etudiants, main = "Effet de la mention au Bac",
        xlab = "Mention au baccalauréat",
        ylab = "Note en MathSV sur 20",
        las = 1, col = "lightblue", varwidth = TRUE,
        notch = TRUE)
```



Donnez le code R permettant de représenter l'effet du statut fumeur on non de la mère sur la masse des bébés.



Réponse 15:

6.3 Deux variables qualitatives

La fonction `table()` permet de prendre en compte simultanémant deux variables qualitatives. Intéressons-nous par exemple au goût pour les mathématiques des étudiants (`gMaths`) et la note qu'ils ont obtenu en mathématiques au baccalauréat (`nMaths`).

```
table(etudiants$gMaths, etudiants$nMaths)
```

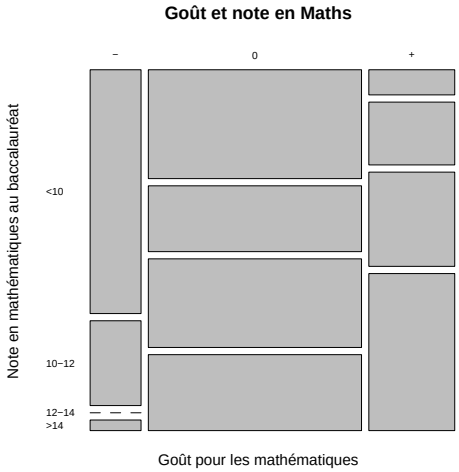
```
   <10 >14 10-12 12-14
-  23   1     8     0
+   4  25    10    15
0  43  30    26    35
```

```
# Ordonnons les modalités :
etudiants$gMaths <- ordered(etudiants$gMaths, levels = c("-", "0", "+"))
etudiants$nMaths <- ordered(etudiants$nMaths, levels = c("<10", "10-12", "12-14", ">14"))
table(etudiants$gMaths, etudiants$nMaths)
```

```
   <10 10-12 12-14 >14
-  23     8     0   1
0  43    26    35  30
+   4    10    15  25
```

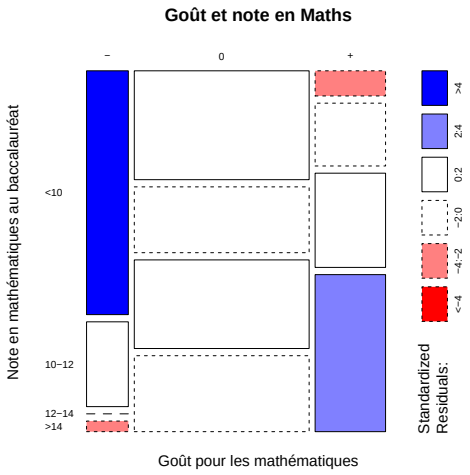
Un tableau tel que celui ci, ventilant des individus entre les modalités croisées de deux variables qualitatives s'appelle une table de contingence. La fonction `mosaicplot()` donne une représentation graphique des tables de contingences :

```
mosaicplot(table(etudiants$gMaths, etudiants$nMaths),
                          main = "Goût et note en Maths",
                          xlab = "Goût pour les mathématiques",
                          ylab = "Note en mathématiques au baccalauréat",
                          las = 1)
```



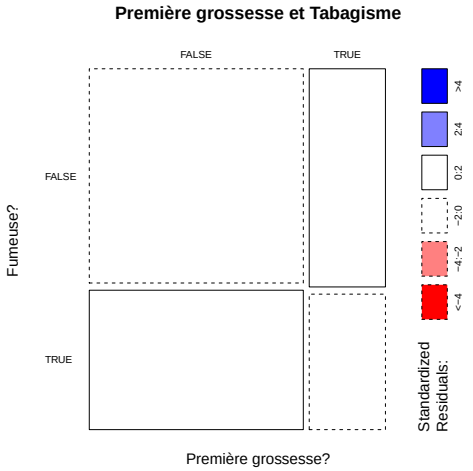
Dans cette représentation la surface des rectangles est proportionnelle aux effectifs des étudiants concernés. Par exemple, le plus grand rectangle ici correspond aux 43 étudiants qui ont un goût modéré pour les mathématiques et une note `<10` en mathématiques au baccalauréat. Une option intéressante de cette fonction est `shade = TRUE` :

```
tabcontin<-table(etudiants$gMaths, etudiants$nMaths)
mosaicplot(tabcontin,
                        main = "Goût et note en Maths",
                        xlab = "Goût pour les mathématiques",
                        ylab = "Note en mathématiques au baccalauréat",
                        las = 1, shade = TRUE)
```



Les graphiques de la fonction `mosaicplot()` seront fort utiles pour interpréter le résultat du test du χ^2 d'indépendance, nous y reviendrons dans le fascicule 2.

Donnez le code R permettant de représenter la relation entre le fait de fumer ou non pour les mères et le fait que cela soit leur première grossesse ou non :



Réponse 16:

7 Visualisation de trois variables à la fois

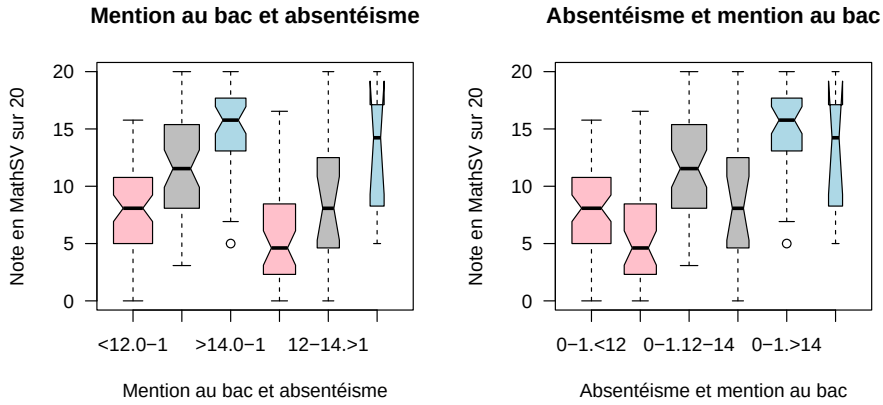
7.1 Quanti-Quali-Quali

On cherche dans ce cas à analyser l'effet de deux variables qualitatives sur une variable quantitative, par exemple, quel est l'effet de la mention au baccalauréat et de l'absentéisme en TD sur la note de MathSV ? La notation `note~mBac+abs` se lit « la note en fonction de la mention au bac et de l'absentéisme ». Le graphique produit va dépendre de l'ordre des variables explicatives. Pour faciliter la lecture on va colorier en rose les mentions `<12`, en gris les mentions `12-14` et en bleu les mentions `>14`.

```
# Ordonnons les modalités d'absentéisme
etudiants$abs <- ordered(etudiants$abs, levels = c("0-1", ">1"))
tapply(etudiants$note, list(etudiants$mBac,etudiants$abs), median)
```

```
       0-1     >1
<12    8.08  4.615
12-14 11.54  8.075
>14   15.77 14.230
```

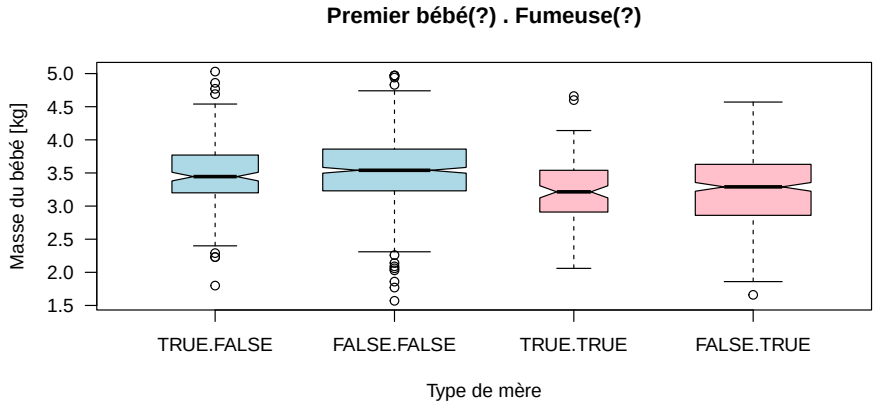
```
par(mfrow = c(1, 2)) # Pour avoir deux graphiques côte à côte
boxplot(note~mBac+abs, etudiants, main = "Mention au bac et absentéisme",
       xlab = "Mention au bac et absentéisme",
       ylab = "Note en MathSV sur 20",
       las = 1, col = c("pink", "grey", "lightblue"),
       varwidth = TRUE, notch = TRUE)
boxplot(note~abs+mBac, etudiants, main = "Absentéisme et mention au bac",
       xlab = "Absentéisme et mention au bac",
       ylab = "Note en MathSV sur 20",
       las = 1, col = rep(c("pink", "grey", "lightblue"), each = 2),
       varwidth = TRUE, notch = TRUE)
```

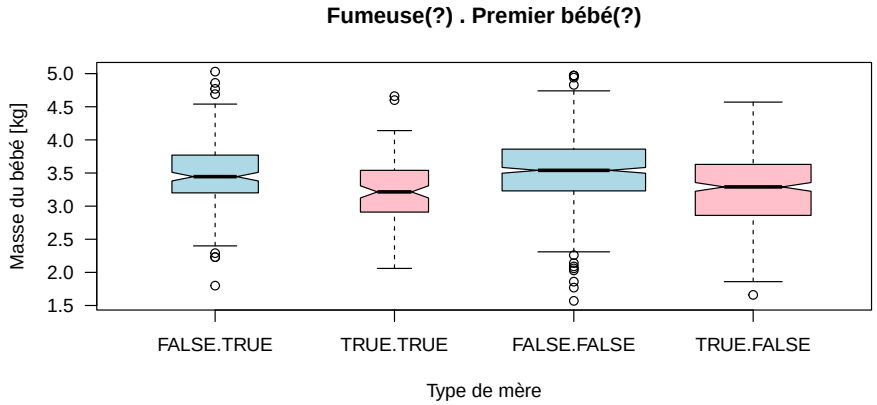


Le graphique de gauche permet de visualiser facilement l'effet de la mention au baccalauréat pour les deux modalités d'absentéisme, le graphique de droite celui de l'absentéisme pour chaque modalité de mention au baccalauréat. Pour pouvoir renseigner la fonction `boxplot()`, il faut avoir une idée précise de ce qu'on veut obtenir comme graphique: si on veut comparer prioritairement les étudiants selon leur mention au bac alors il faut placer la variable `mBac` en premier dans la parenthèse, et si on veut plutôt évaluer d'abord l'effet de l'absentéisme sur les notes obtenues en mathsv, il convient de mettre la variable `abs` en premier).

Donnez le code R permettant de représenter l'effet simultané du tabagisme et de la parité sur la masse des bébés à la naissance. Pour faciliter la lecture des graphiques on pourra utiliser le recodage suivant :

```
bebes$smoke <- ordered(ifelse(bebes$smoke, TRUE, FALSE), levels = c(FALSE, TRUE))
bebes$parity <- ordered(ifelse(bebes$parity, TRUE, FALSE), levels = c(TRUE, FALSE))
```





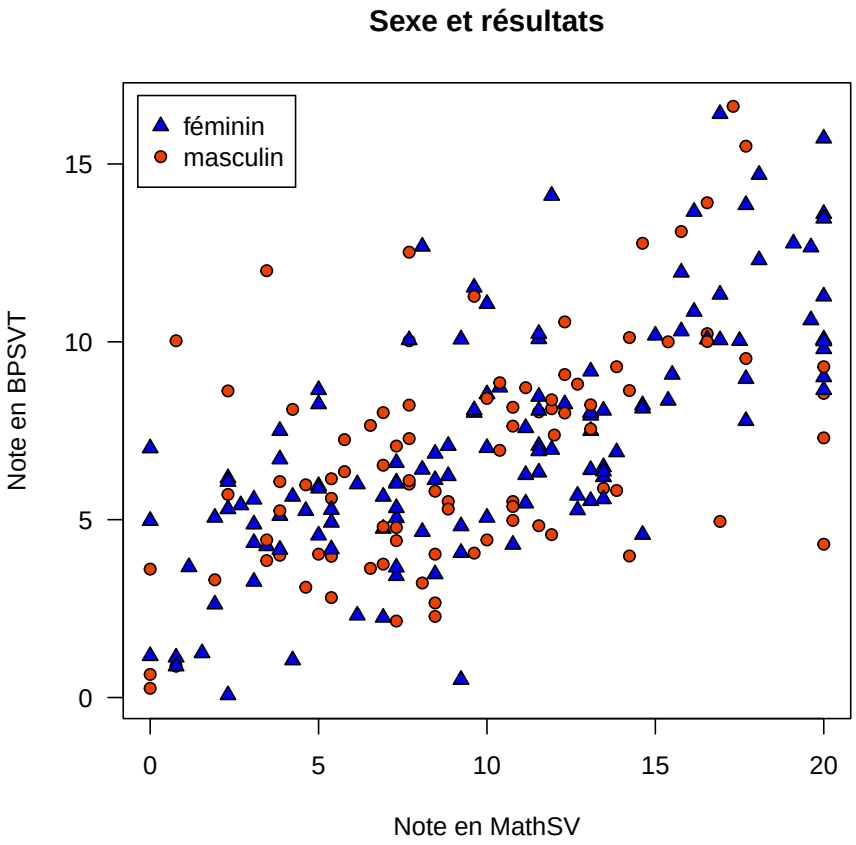
Réponse 17:

Le graphique fourni ci-dessus pourra être utile notamment pour comprendre les résultats d'une analyse de la variance à deux facteurs contrôlés.

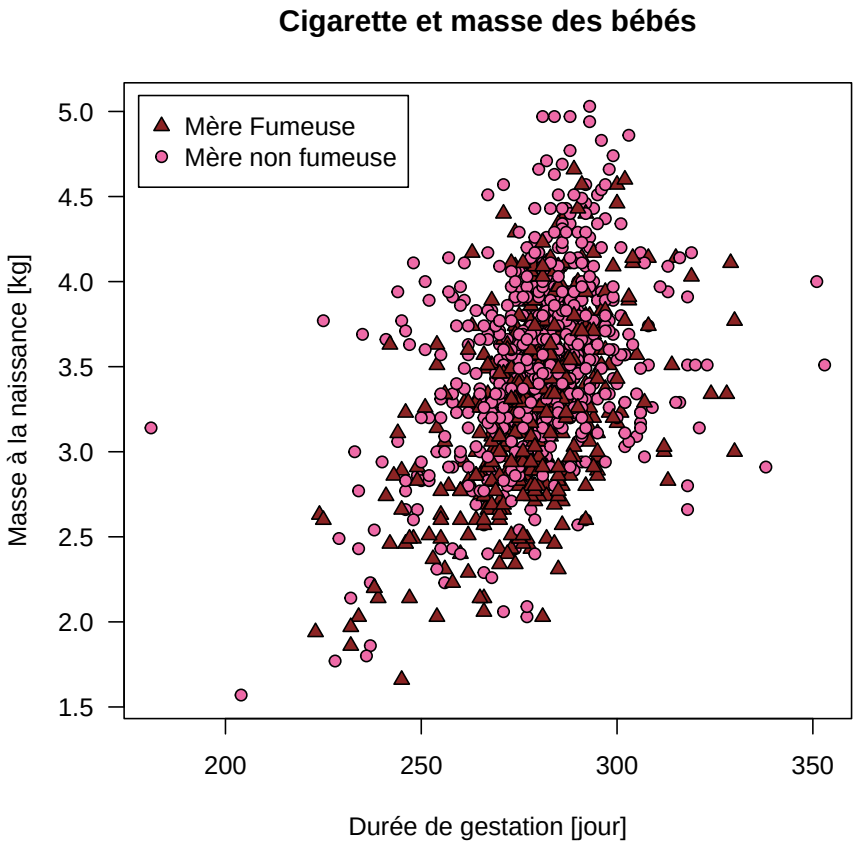
7.2 Quanti-Quanti-Quali (Pour aller plus loin)

Il s'agit ici d'un nuage de points sur lequel on porte une information supplémentaire à l'aide d'un code graphique (type et couleur des points). La liste des types de points disponibles est donnée dans la documentation de la fonction `points()`. Un aperçu des couleurs pré-définies est donné en invoquant `demo("colors")` dans la console. Par exemple, on aimerait connaître le sexe des étudiants quand on croise la note en MathSV et la note en BPSVT :

```
with(etudiants,
      plot(note, nBPSVT, las = 1,
      xlab = "Note en MathSV", ylab = "Note en BPSVT",
      pch = ifelse(sexe == "F", 24, 21),
      main = "Sexe et résultats",
      bg = ifelse(sexe == "F", "blue2", "orangered2")))
#
legend("topleft", inset = 0.02, legend = c("féminin", "masculin"), pch = c(24, 21),
       pt.bg = c("blue2", "orangered2"))
```



Donnez le code R permettant de distinguer graphiquement les mères fumeuses et non fumeuses quand on croise le temps de gestation avec la masse à la naissance des bébés :



Réponse 18:

8 Sauvegarde des données

Vous allez utiliser les données `etudiants` et `bebes` lors du prochain TP. Pour sauvegarder d'un coup tous les objets définis dans votre environnement cliquez sur l'icône qui ressemble à une disquette dans l'onglet « *Environment* » en haut à droite de l'interface de `RStudio`. Pour restaurer tout votre environnement à la prochaine séance il suffira de cliquer sur l'icône juste à gauche de la précédente (elle représente une flèche qui sort d'un dossier).