TP R Introduction à la statistique

Mohamed FOFANA | *L1 MIASHS - Université Grenoble Alpes*

Contents

1 TP 1: Statistique descriptives - Boîtes de distribution

1.1 Prise en main de R

1.1.1 R une calculette scientifique

R permet de faire des opérations de calcul élémentaire. Essayer les commandes suivantes:

```
calculs = c(
4*5-2^4,
1+14/4
)
calculs
```

```
## [1] 4.0 4.5
```

R permet de faire des caculs plus élaborés. Il utilise pour cela des fonctions. Plusieurs fonction prédéfinies sont disponibles. Que font les fonction suivantes ?

```
fonction = c(
sqrt = sqrt(5),
abs = abs(-4),
log = log(1),
co = cos(pi),
exp = exp(1),
round = round(pi,2))
fonction
```

```
##      sqrt       abs       log        co       exp     round
##  2.236068  4.000000  0.000000 -1.000000  2.718282  3.140000
```

1.1.2 R est un language de programmation

```
variables = c(
x<-2.5,
x = x,
y<-2*x,
y1= y,
```

```
y<-1+log(y)^2,
y2 = y,
y<-floor(y),  # partie entière de x
y3 = y)
variables
```

1.1.2.1 Création de variables

```
##                x              y1              y2              y3
## 2.50000 2.50000 5.00000 5.00000 3.59029 3.59029 3.00000 3.00000
```

Dand la première ligne, On a créé une variable de nom x contenant la valeur 2.5. la flèche, obtenu en tapant < et - est appelée opérateur d'affectation. Cet opérateur dit que la variable x doit contenir la valeur 2.5. Si cette variable n'existe pas, elle est créée? Si elle existe, son contenu sera remplacé par la nouvelle valeur.Pour rappeler le contenu d'une variable, il suffit de taper son nom(ligne 2)

1.1.2.2 Création d'un vecteur Il ya différentes manières de créer des vecteurs dans R. Etudiez les ligne de commande suivantes

```
vec<-c(-1,5,2,0.5,-5)
vec
```

```
## [1] -1.0  5.0  2.0  0.5 -5.0
```

```
vec1<-c(3,10,vec)
vec1
```

```
## [1]  3.0 10.0 -1.0  5.0  2.0  0.5 -5.0
```

```
length(vec1)
```

```
## [1] 7
```

```
rep(c(3,5),2)
```

```
## [1] 3 5 3 5
```

```
1:20
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
```

```
5:1
```

```
## [1] 5 4 3 2 1
```

```
seq(from=1, to=16, by=3)
```

```
## [1]  1  4  7 10 13 16
```

```
seq(from=1, to=2, length=10)
```

```
##  [1] 1.000000 1.111111 1.222222 1.333333 1.444444 1.555556 1.666667 1.777778
##  [9] 1.888889 2.000000
```

1.1.2.3 Opérations sur les vecteurs Créer deux vecteurs x et y quelconques et de même taille. Que font les commandes suivantes?

```
vecteurs= c(
sumprod = 2*x+1,
exp = exp(y),
addition = x+y,
multiplication = x*y,
```

```
somme_des_colonne = sum(x),
consum = cumsum(x),
which = which(y<0),
which.min = which.min(y),
sort = sort(y))
vecteurs
```

```
##           sumprod               exp          addition    multiplication
##           6.00000          20.08554           5.50000           7.50000
## somme_des_colonne            consum         which.min              sort
##           2.50000           2.50000           1.00000           3.00000
```

1.1.2.4 Accès aux éléments d'un vecteur On peut accéder aux éléments d'un vecteur grâce à l'opérateur "[]".

```
x<-c(3,-1,5,7,0,3,2,-9)
y<-c("rouge","noir","vert") # vecteur de chaines de charactères
x[1:3]
```

```
## [1]  3 -1  5
```

```
x[-1]
```

```
## [1] -1  5  7  0  3  2 -9
```

```
z=y[y!="rouge"]
z
```

```
## [1] "noir" "vert"
```

1.2 R pour les Statistiques

1.2.1 Les dataFrames

Tout comme les vecteurs, les dataframes sont des mais plus élaborés et spécialement désignés pour le stockage de données. Il s'agit plus ou moins d'un tableau à 2 dimensions (une matrice), mais avec plus des noms de colonnes, des noms de lignes, etc. Très généralement, et ce sera toujours le cas en ce qui nous concerne, les ligne d'un dataframes seront des individus(dans l'exemple cidessous ce sont des sujets) et les colonnes des variables(dans l'exemple ci-dessous ce sont le poids, la tailes et l'indice de Masse Corporelle des sujets).

```
poids = c(65,82,45,63,70)
taille=c(1.75,1,78,1.52,1.57,1.80)
IMC = poids/(taille^2)  # IMC est l'indice de masse corporelle
```

```
## Warning in poids/(taille^2): la taille d'un objet plus long n'est pas multiple
## de la taille d'un objet plus court
```

```
data = cbind(poids,taille,IMC)
```

```
## Warning in cbind(poids, taille, IMC): number of rows of result is not a
## multiple of vector length (arg 1)
```

```
# cbind permet de coller des colonnes les unes à la suite de l'autres
# rbind permet de coller des lignes les unes à la suite de l'autres
```

De la mêmes maniere que pour les vecteurs, les crochets permettent d'aller chercher des éléments d'une matrice ou d'un tablea. Il y'a alors 2 paramètres à définir, le premier pour les lignes et le second pour les colonne à sélectionner(les 2 séparés par une virgule) Si rien n'est préciser pour les premier paramètre, cela

signifie que l'on prend toutes les ligne. Si rien n'est préciser pour les deuxième paramètre, cela signifie que l'on prend toutes les colonnes.

```
data[3,2] # taille du troisième sujet
```

```
## taille
##     78
```

```
data[,1]  # taille de tous les sujets
```

```
## [1] 65 82 45 63 70 65
```

```
d=c(
dim = dim(data),
nrow = nrow(data),
ncol = ncol(data),
rownames = rownames(data),
colnames = colnames(data))
d
```

```
##      dim1      dim2      nrow      ncol colnames1 colnames2 colnames3
##       "6"       "3"       "6"       "3"   "poids"  "taille"     "IMC"
```

1.2.2 Les fonctions statistiques de R

On considère la serie statistique

```
x=c(1,2,4,8,9,15,6,8,18,7,5,2,4,6,8,9)
```

R possède plusieurs fonction statistique. Que font les foction suivantes ?

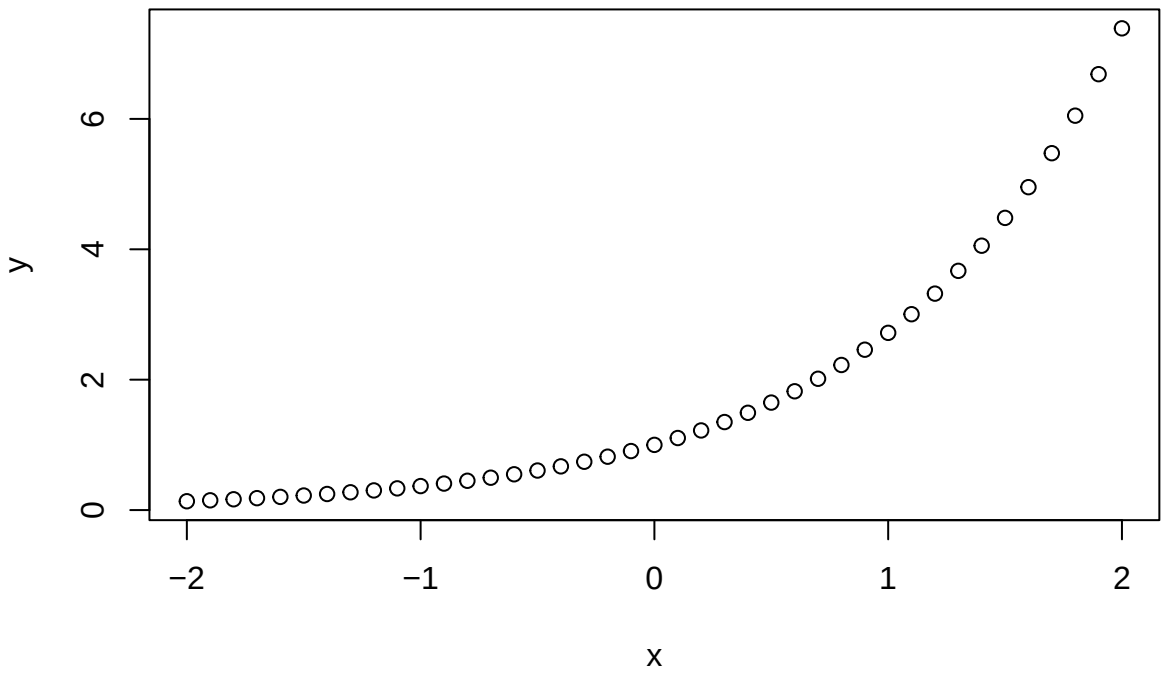
```
b= c(
min = min(x),
max = max(x),
mean = mean(x),
sd = sd(x),
quantile = quantile(x,0.25),
median = median(x),
var = var(x),
summary(x)
)
b
```

```
##          min          max         mean           sd quantile.25%       median
##     1.000000    18.000000     7.000000     4.516636     4.000000     6.500000
##          var         Min.      1st Qu.       Median         Mean      3rd Qu.
##    20.400000     1.000000     4.000000     6.500000     7.000000     8.250000
##         Max.
##    18.000000
```

1.2.3 Représentations graphiques

La fonction `plot` est la fonction générique de graphe

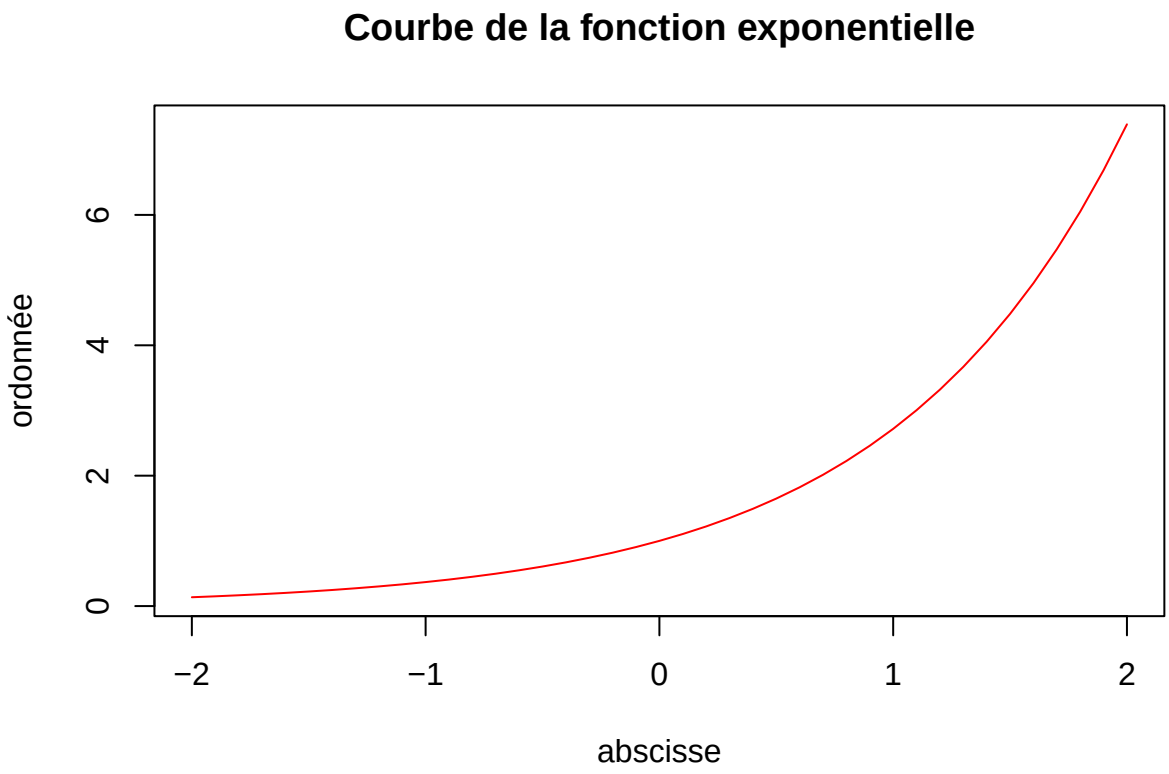
```
x=seq(-2,2,by=0.1)
y=exp(x)
plot(x,y)
```



Il est possible de parametrer l'affichage obtenu(couleurs,etc). Voici un bref apreçu des options de la fonction plot, vous pouvez regarder l'aide(en tapant `?plot`) pour plus de details. Nous aurons de toutes les façon l'occasion de revenir sur l'utilisation de cette fonction.

- **type** (chaine de charactères) "p" pour points, "l" pour lignes, "b" pour les deux (both);
- **main** (chaine de charactères) un titre pour le graphique;
- **sub** (chaine de charactères) un sous-titre pour le graphique;
- **xlab** (chaine de charactères) un titre pour l'axe des abscisses;
- **ylab** (chaine de charactères) un titre pour l'axe des ordonnées;
- **col** (vecteur de chaines de charactères) la ou les couleurs(s) à utiliser;
- **etc...**

```
plot(x,y,pch = 1, type="l", col="red", main = "Courbe de la fonction exponentielle", xlab="abscisse"
, ylab= "ordonnée" )
```



1.3 Criminalité - Peine de mort

Le but de l'étude statistique de cette partie est de s'éclairer sur le lien potentiel entre abolition de la peine de mort et taux de criminalité. Nous resterons néanmoins très prudent sur les conclusions que l'on peut rendre avec des outils de statistiques descriptives. Nous vous conseillons dès à présent d'allumer votre ordinateur et de lancer le logiciel R.

En annexe 1, on trouvera des données issues du site du FBI. Elle donnent le nombre de meurtres dans les différents états des Etats Unis en 2009 ainsi que la population. Afin de s'affranchir des différences de tailles de populations dans chaque état, nous contruisonsle taux de criminalité pour 10 000 habitants.

J'ai malheureusement renversé mon café sur la colonne ***taux de criminalité***.

1.3.1 Construction de la variable taux de criminalité pour les états ayant aboli la peine de mort

(i) A partir des colonnes 2 et 3 de l'annexe, comment calculer le taux de criminalité pour 10 000 habitants ?

(ii) Nous allons créer la variable série statistique abolie09 contenant le taux de criminalité pour 10 000 habitants des états ayant aboli la peine de mort. Pour cela, nous allons saisir les variables **murder** et **population**. Nous calculerons alors la variable **abolie09.**

```
murder=c(7,34,72,9,21,26,144,31,22,319,76,625,144)
population=c(622,3008,5266,647,1395,1318,5655,1053,698,8708,1820,9970,2010)
```

On rappelle que dans R, une série statistique est un vecteur horizontal. On peut affecter une valeur à une variable en utilisant au choix <- ou =.

Calculons la variable abolie09 en divisant terme à terme murder par population et en multipliant par 10 (pour un taux par dizaine de milliers d'habitants).

```
abolie09=10*murder/population
```

On pourra afficher en tapant *abolie09 puis entrée* pour vérifier l'adéquation avec ce qui reste visible dans le tableau en annexe.

1.3.2 Comparaison des variables taux de criminalité pour les états ayant aboli ou non la peine de mort

Nous avons en mémoire la variable abolie09. Je vous propose de mettre en mémoire la variable nonabolie09 qui porte le taux de criminalité dans les pays n'ayant pas aboli la peine de mort en 2009.

```
nonabolie09=c(0.075,0.133,0.135,0.142,0.202,0.217,0.223,0.254,0.256,0.287,0.304,
0.332,0.371,0.394,0.399,0.419,0.435,0.440,0.456,0.463,0.497,0.512,0.512,0.522,
0.534,0.535,0.552,0.590,0.592,0.610,0.627,0.636,0.675,0.732,0.768,1.082,2.401)
```

Vous noterez l'indispensable d'utiliser d'autres méthodes d'acquisition des données pour les grandes séries. Nous pouvons vérifier qu'il ne manque pas de valeur en demandant le nombres de valeurs dans la série tapée :

```
length(nonabolie09)
```

```
## [1] 37
```

(i) A l'aide de la fonction `mean()` calculer les moyennes de **abolie09 et nonabolie09**. Comparer. Que peut-on en conclure ?

(ii) On rappelle ce qu'est le diagramme *boîte de distribution*.

La boîte de distribution (ou boxplot) est une représentation graphique synthétique de la distribution des données. Elle résume quelques caractéristiques de position et de dispersion du caractère étudié (médiane, quartiles, minimum et maximum). Ce diagramme est utilisé essentiellement pour comparer un même caractère dans des populations différentes, ou une évolution au cours du temps.

(a) Tracer un rectangle qui s'étend du premier quartile au troisième.

(b) Séparer ce rectangle en deux à la hauteur de la médiane. On obtient alors une boîte.

(c) On complète ce rectangles par deux segments. Pour cela, on calcule

$$a = q_{0.25} - 1.5IQ \quad \text{et} \quad b = q_{0.75} + 1.5IQ,$$

avec la distance inter-quartile

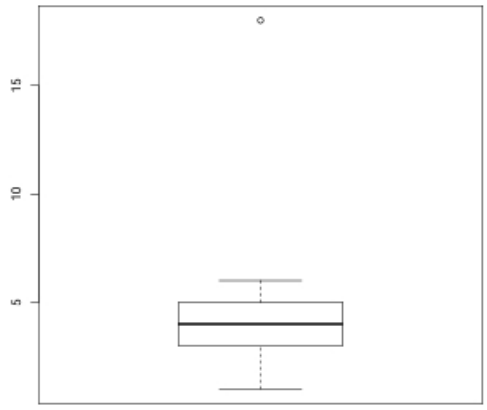
$$IQ = q_{0.75} - q_{0.25}.$$

On repère les valeurs :

$$x_a = \min\{x_i : x_i \geq a\} \quad \text{et} \quad x_b = \max\{x_i : x_i \leq b\}.$$

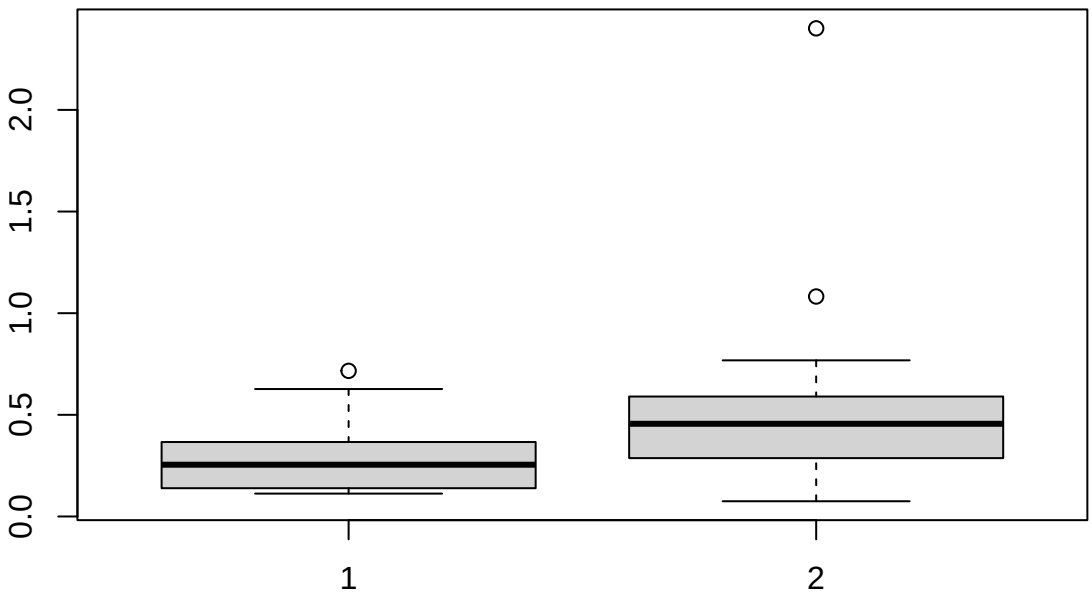
Ces valeurs sont appelées *valeurs adjacentes*. On relie ces valeurs aux cotés de la boîte.

(d) Les valeurs qui ne sont pas comprises entre les valeurs adjacentes sont représentées par des points et sont appelées *valeurs extrêmes*.



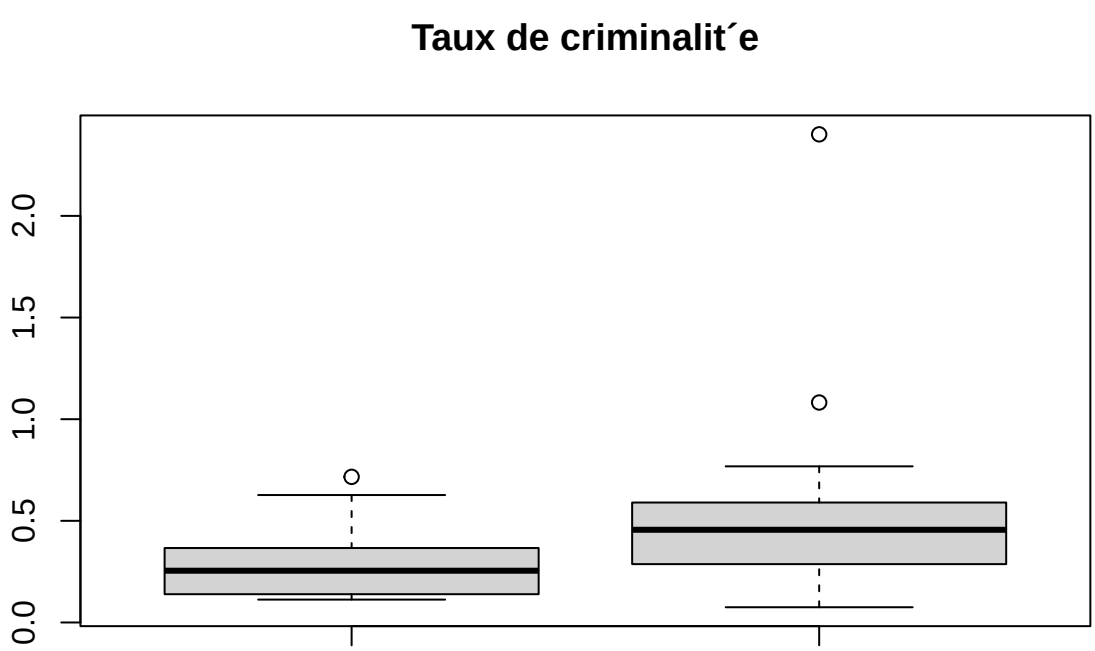
On va tracer les boîtes de distribution des deux séries **abolie09** et **nonabolie09** côte à côte pour avoir une meilleur vision de leur distribution.

```
boxplot(abolie09,nonabolie09)
```



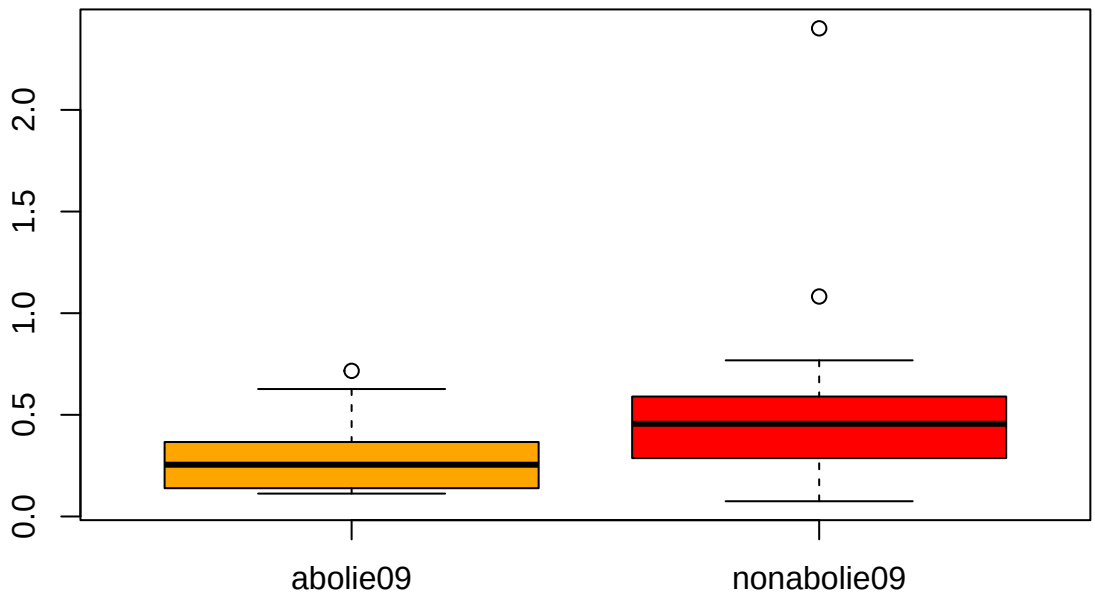
On pourra modifier les paramètres du graphe, par exemple ajouter un titre principal avec le paramètre main :

```
boxplot(abolie09,nonabolie09,main="Taux de criminalit´e")
```



Les accents sont pris en charge. Les chaines de caractère doivent être mises entre guillemets. On peut également changer le nom de chaque variable et sa couleur avec les paramètres **names** et **col** dont les arguments sont des vecteurs portant la caractéristique souhaitée pour chaque série dans l'ordre.

```
boxplot(abolie09,nonabolie09,main="Taux de criminalit´e",col=c("orange","red"),
 names=c("abolie09","nonabolie09"))
```



Que peut-on déduire de ce graphe ?

(iii) Peut-on déduire un lien de cause à effet avec ce type détude ?

1.3.3 Autre démarche pour obtenir le boxplot

Supposons que, comme c'est dans la présentation de l'annexe, les deux séries **abolie09 et nonabolie09** soient concaténées dans un seul vecteur de données

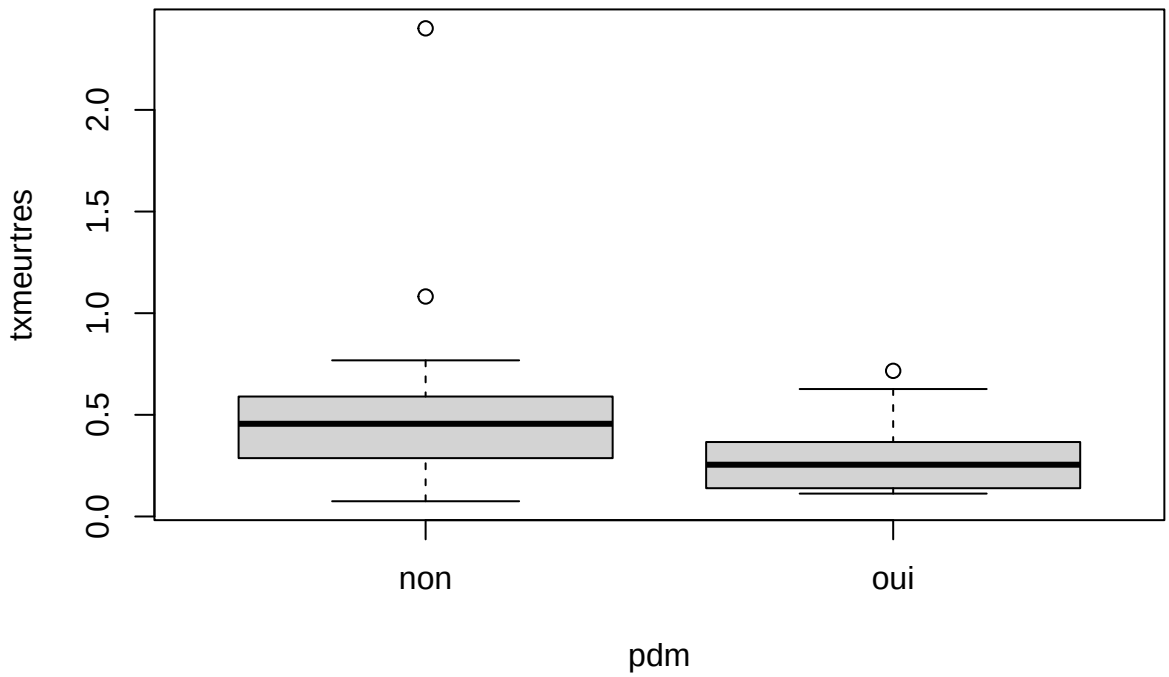
```
txmeurtres=c(abolie09,nonabolie09)
```

Faites afficher **txmeurtres** pour vous en persuader. On peut créer la variable qui discrimine l'abolition ou non de la peine de mort.

```
pdm=c(rep("oui",13),rep("non",37))
```

Faites afficher pdm pour vérifier que nous avons le statut de l'abolition ou non de la peine de mort pour chaque état dans le même ordre que le taux de criminalité. On pourra obtenir le même boxplot que précdédemment en discriminant les donn´es sur le facteur pdm.

```
boxplot(txmeurtres~pdm)
```


txmeurtres
0.0
0.5
1.0
1.5
2.0
non
oui
pdm