

Statistiques avec R

SCI 1018

Les graphiques avec R

Marc J. Mazerolle

Département des sciences du bois et de la forêt, Université Laval



© TÉLUQ, 2019

Table des matières

1	Graphiques de base	2
2	Histogramme	2
3	Diagramme à bâtons	2
4	Diagramme de boîtes et moustaches	2
5	Graphiques génériques	3
6	Sauvegarder des graphiques	3
	Index	12

1 Graphiques de base

R a été conçu pour faire des analyses statistiques et des graphiques. Ce n'est donc pas surprenant d'y trouver plusieurs fonctions graphiques, des plus simples aux plus élaborées. Pour l'instant, passons en revue les fonctions graphiques de base.

2 Histogramme

La fonction `hist()` permet de construire un histogramme de fréquence à partir d'une variable numérique (fig. 1). On remarque que les étiquettes de l'axe des x's (abscisses) et des y's (ordonnées) ainsi que le titre principal laissent un peu à désirer : par défaut, `hist()` affiche le nom de la variable tel qu'il a été spécifié. On peut changer les étiquettes à l'aide des arguments `xlab`, `ylab` et `main` (fig. 2).

3 Diagramme à bâtons

Pour tracer un diagramme à bâtons, typiquement nécessaire pour illustrer des valeurs discrètes, on utilise `barplot()` (fig. 3). Cette fonction nécessite un vecteur ou une matrice comme argument principal. Quant aux matrices, on peut soit illustrer les résultats sous forme de bâtons superposés ou adjacents (fig. 4).

On remarque qu'une légende a été ajoutée à l'aide de l'argument `legend.text = TRUE` (fig. 4). Nous verrons plus tard que la fonction `legend()` est beaucoup plus flexible. Il est à noter également que l'ajout de couleur peut se faire directement avec l'argument `col`.

4 Diagramme de boîtes et moustaches

On peut obtenir un diagramme de boîtes et moustaches (*boxplot* ou *box-and-whisker plot*) à l'aide de la fonction `boxplot()` (fig. 5). Cette fonction permet d'illustrer une seule variable ou de visualiser plusieurs groupes simultanément. Ce graphique illustre la médiane (et non

la moyenne arithmétique) par un trait gras à l'intérieur de la boîte. La limite inférieure de la boîte correspond au 25^e percentile tandis que la limite supérieure indique le 75^e percentile. La différence entre ces deux limites se nomme l'écart interquartile. Le graphique de boîtes et moustaches montre où se situent 50% des observations et s'avère un bon moyen de visualiser rapidement les données. Par défaut, les moustaches représentent 1.5 fois l'écart interquartile ou la valeur la plus extrême : la moustache présente la plus petite de ces deux valeurs. Ceci peut-être modifié à l'aide de l'argument `range`. Par exemple, si `range = 0`, les moustaches s'étendront jusqu'aux valeurs extrêmes.

5 Graphiques génériques

La fonction `plot()` permet de créer des nuages de points (`type = "p"`), des lignes (`type = "l"`), une combinaison de lignes et de points (`type = "b"`), et quelques autres graphiques inusités (fig. 6). Cette fonction comporte un grand nombre d'arguments. On peut modifier plusieurs paramètres graphiques généraux de R à l'aide de `par()`. Par exemple, on peut ajouter plusieurs graphiques dans une même fenêtre, à l'aide des arguments `mfrow` ou `mfcop`. La première remplit la fenêtre une rangée à la fois, alors que la deuxième remplit la fenêtre une colonne à la fois.

Un point important pour créer des lignes : il faut ordonner les données selon la variable indépendante (i.e., l'axe des x's), sinon on obtient des zigzags qui n'ont aucun sens (fig. 6).

6 Sauvegarder des graphiques

Lorsqu'on a obtenu le graphique convoité, on peut le sauvegarder facilement. Plusieurs options se présentent. Par exemple, sous MS-Windows, on peut copier le graphique à partir de la fenêtre graphique et le coller directement dans un document MS-Word ou `Writer` de `OpenOffice`. Le format vectoriel (*metafile*) est un format particulièrement facile à coller directement dans les documents. Autrement, on peut le sauvegarder sous différents formats à

partir du menu dans la fenêtre du graphique nouvellement créé. Une autre option encore plus flexible consiste à utiliser les appareils graphiques (*graphical devices*) afin de spécifier le fichier dans lequel le graphique sera sauvegardé, et de spécifier plusieurs caractéristiques, comme la résolution désirée ou la taille en pixels, en centimètres ou en pouces. Une vérification rapide avec `?Devices` nous indique qu'on peut sauvegarder entre autres, en format pdf, bitmap, postscript, jpeg, png, et tiff. Pour sauvegarder un diagramme de boîtes et moustaches en pdf, on pourrait procéder ainsi :

```
> ##on indique à R qu'on veut sauvegarder en pdf
> pdf(file = "boxplot.pdf", width = 6, height = 6)
> ##on crée le graphique - n'apparaîtra pas à l'écran
> boxplot(Densite.vers ~ Vegetation, data = vers)
> ##on indique à R que le graphique est terminé
> dev.off( ) #

pdf
2
```

On remarque que la commande `pdf()` indique que le graphique sera sauvegardé sous le format pdf dans le fichier `boxplot.pdf` à l'endroit spécifié. Si aucun chemin n'est spécifié, le fichier sera sauvegardé dans le répertoire de travail. On peut facilement déterminer le répertoire travail à l'aide de la commande `getwd()`. Les arguments `width` et `height` de `pdf()` spécifient la largeur et la hauteur de la figure en pouces, et ces valeurs prennent 7 par défaut. La commande `dev.off()` émise après le graphique informe R que le graphique est terminé. Le graphique à réaliser doit se trouver entre ces deux commandes et n'apparaîtra pas à l'écran puisqu'il sera sauvegardé dans le fichier directement.

De façon similaire, on peut créer un fichier au format jpeg avec la commande suivante :

```
> ##on indique à R qu'on veut sauvegarder en jpeg
> jpeg(file = "boxplot.jpeg", width = 1200,
```

```
      height = 1200, quality = 100)
> ##on crée le graphique - n'apparaîtra pas à l'écran
> boxplot(Densite.vers ~ Vegetation, data = vers)
> ##on indique à R que le graphique est terminé
> dev.off( )

pdf

2
```

Le graphique sera sauvegardé en format jpeg à l'endroit spécifié avec une taille de 1200 x 1200 pixels et une qualité de 100 % (avec peu de compression). Nous continuerons à explorer d'autres fonctionnalités graphiques lors des leçons subséquentes.

```
> #à partir du jeu de données vers  
> vers <- read.table(file = "vers.txt", header = TRUE)  
> hist(vers$Superficie)
```

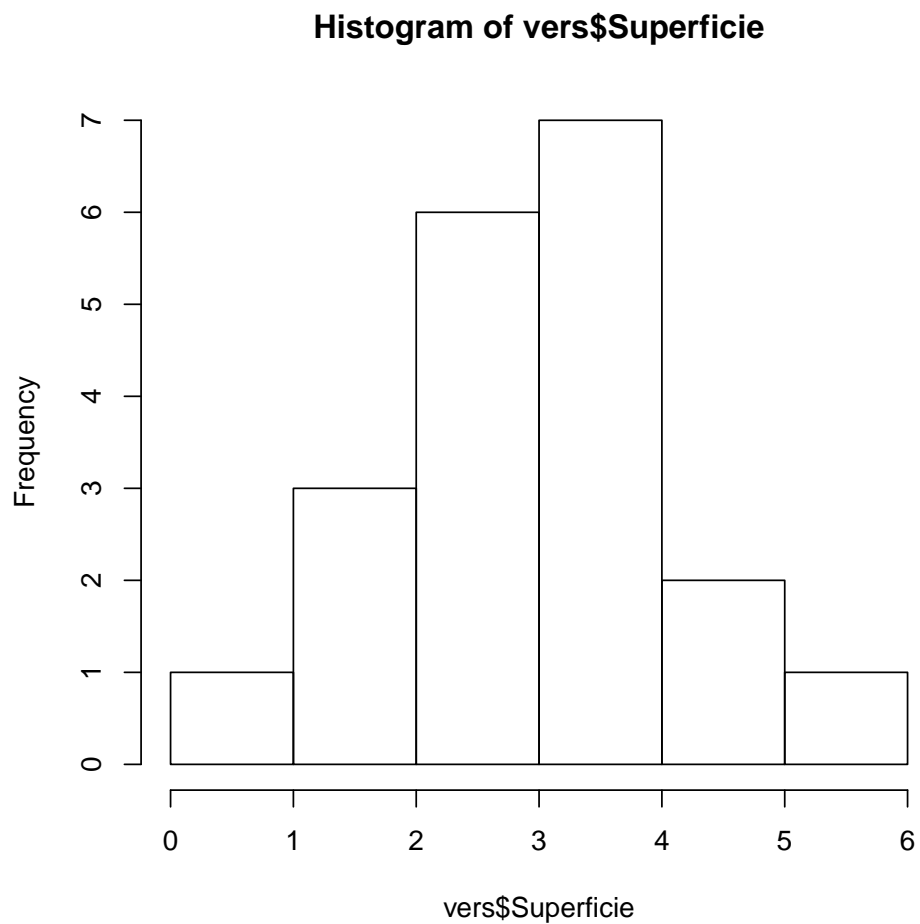


FIGURE 1 – Histogramme des valeurs de superficie dans le jeu de données de vers de terre.

```
> hist(vers$Superficie, ylab = "Fréquences", xlab = "Superficie",  
      main = "Histogramme de fréquences pour les valeurs de superficie")
```

Histogramme de fréquences pour les valeurs de superficie

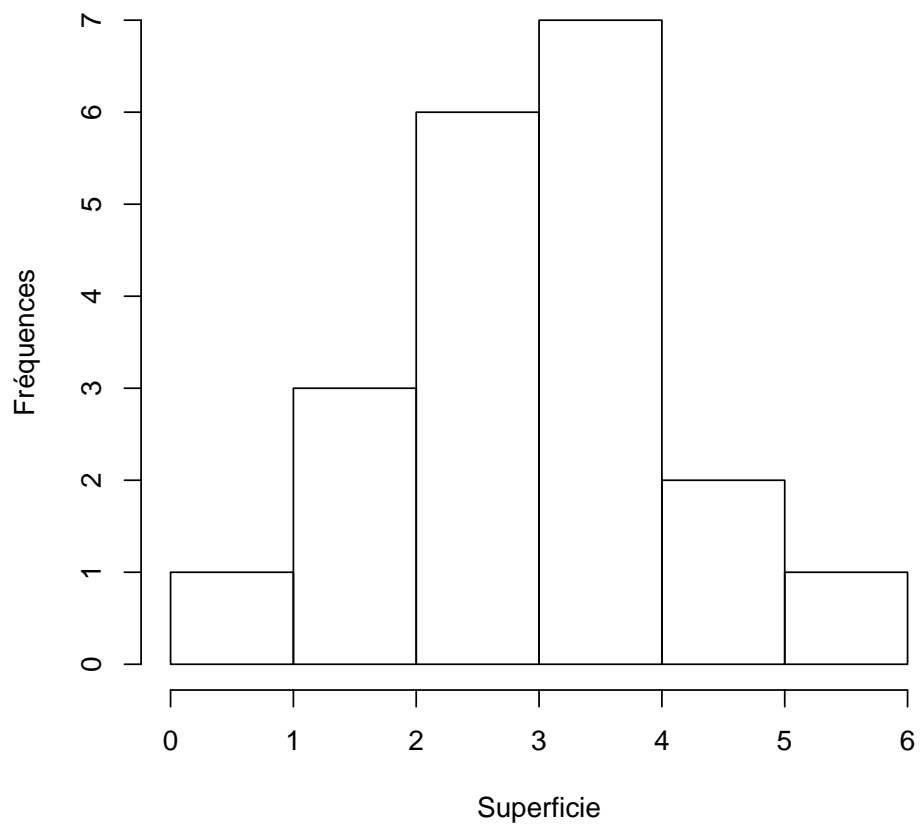


FIGURE 2 – Histogramme plus élaboré à partir des valeurs de superficie dans le jeu de données de vers de terre.


```
> table(vers$Vegetation) #fréquences de chaque type de végétation
  Arable Chaparral  Prairie      Pre  Verger
      3         4         9         3      1
> barplot(height = table(vers$Vegetation), ylab = "Fréquences",
          xlab = "Type de végétation", col = "bisque") #diagramme en bâton
```

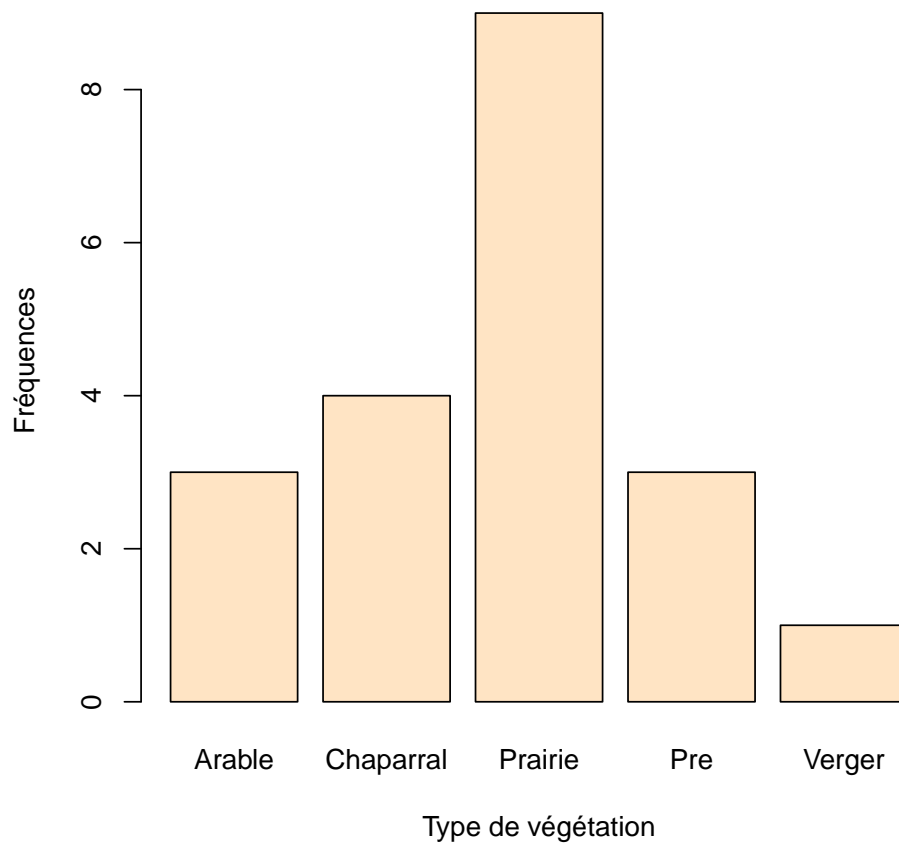


FIGURE 3 – Diagramme à bâtons des fréquences de différentes classes de végétation dans le jeu de données de vers de terre.

```

> tab.freq <- table(vers$Humide, vers$Vegetation)
> tab.freq
      Arable Chaparral Prairie Pre Verger
FALSE      3         2      8  0     1
TRUE       0         2      1  3     0
> par(mfrow = c(2, 2)) #organiser 4 graphiques dans une matrice de 2 x 2
> barplot(tab.freq, main = "Bâtons superposés") #bâtons superposés
> barplot(tab.freq, beside = TRUE,
          main = "Bâtons adjacents") #bâtons adjacents
> barplot(tab.freq, beside = TRUE, legend.text = TRUE,
          main = "Ajouter une légende") #ajouter une légende
> barplot(tab.freq, beside = TRUE, legend.text = TRUE,
          col = c("green", "turquoise"),
          main = "Légende et couleurs") #ajouter légende et couleurs

```

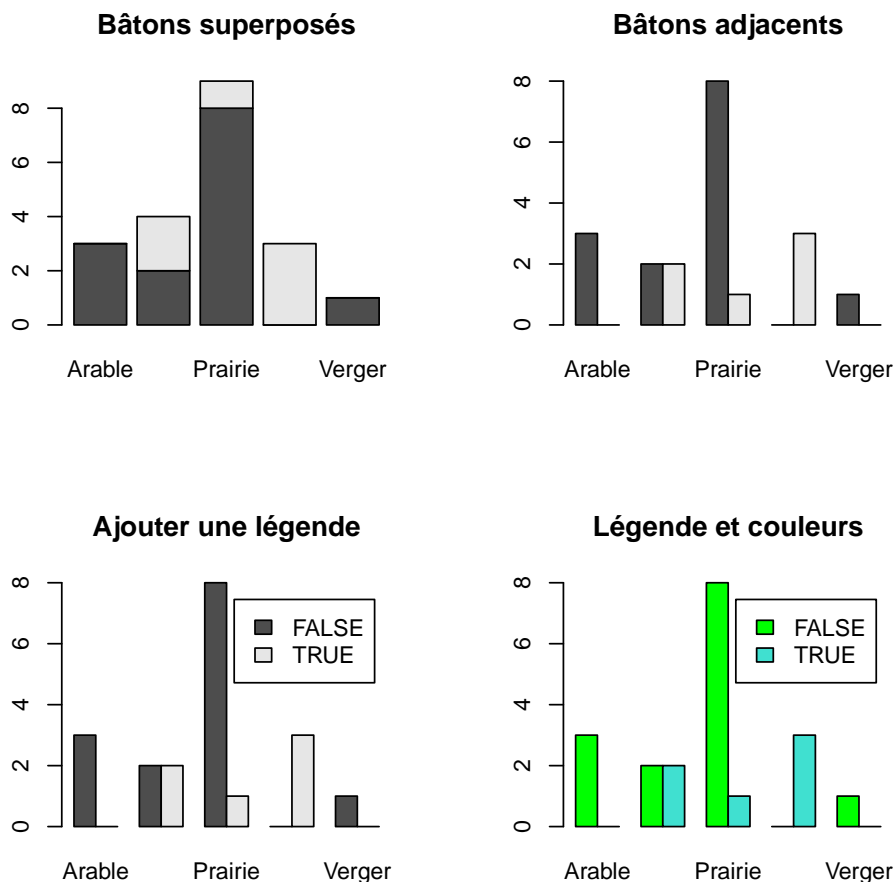


FIGURE 4 – Diagramme à bâtons avec différentes structures et couleurs à partir des fréquences de chaque classe de végétation.

```
> boxplot(Densite.vers ~ Vegetation, data = vers)
```

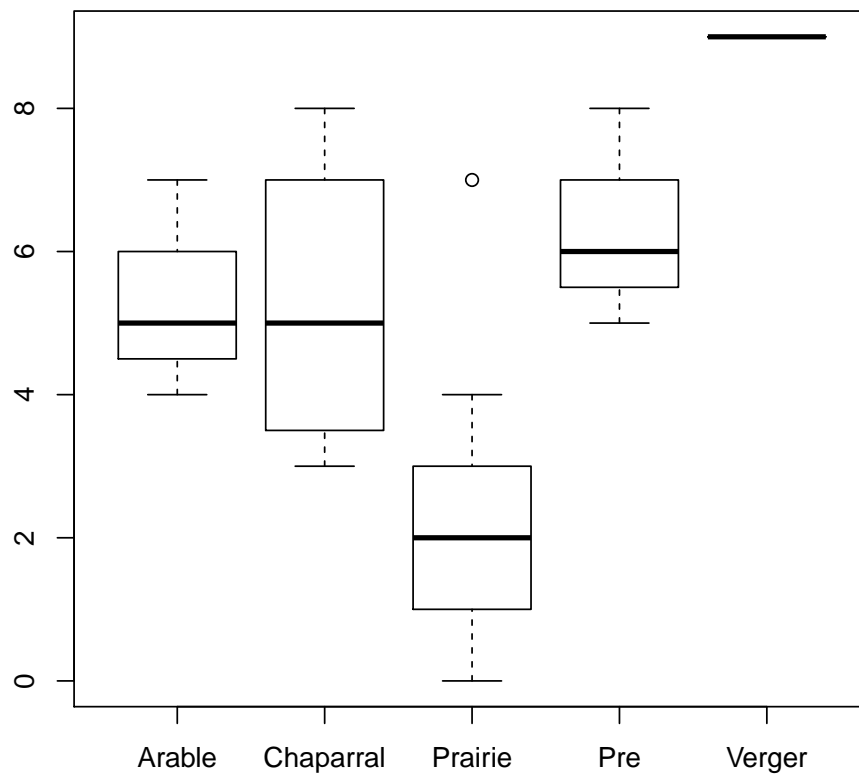


FIGURE 5 – Diagramme de boîtes et moustaches de densités de vers de terre pour chaque classe de végétation.

```

> vers.ord <- vers[order(vers$Superficie), ] #ordonner selon Superficie
> par(mfrow = c(2, 2)) #combiner graphiques sur c(2 rangées, 2 colonnes)
> plot(Densite.vers ~ Superficie, type = "p", data = vers) #points
> plot(Densite.vers ~ Superficie, type = "l", data = vers) #lignes
> plot(Densite.vers ~ Superficie, type = "l", data = vers.ord) #lignes
> plot(Densite.vers ~ Superficie, type = "b", data = vers.ord) #points et lignes

```

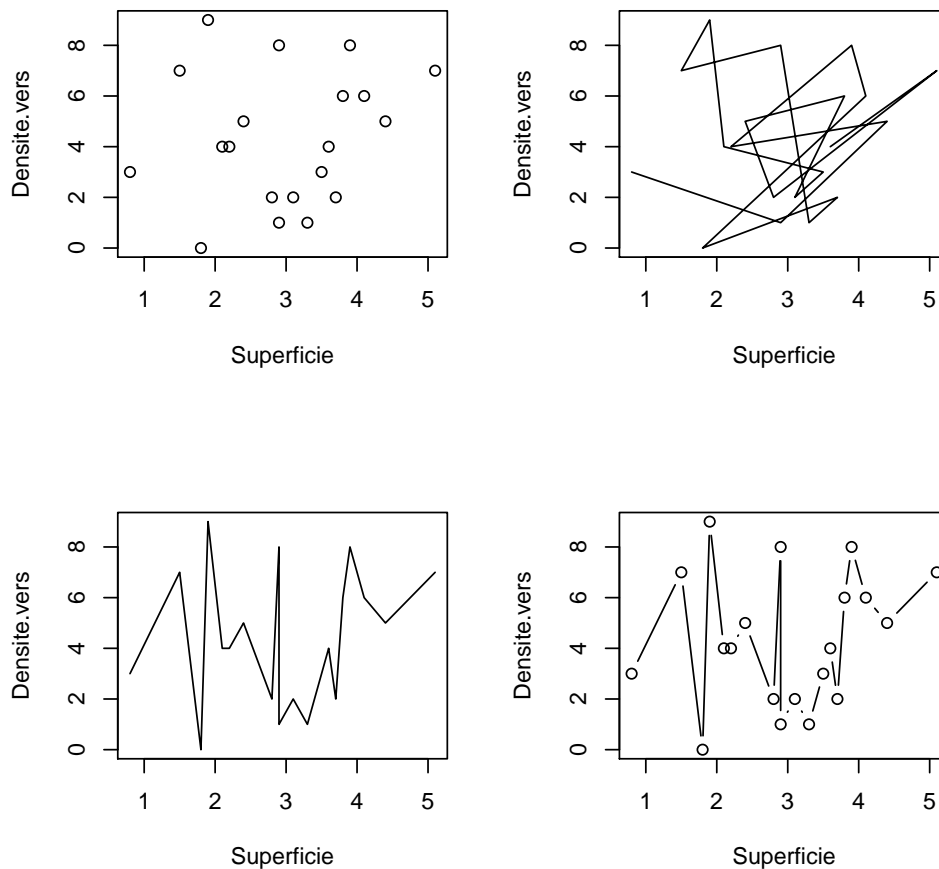


FIGURE 6 – Différents types de graphiques obtenus avec la fonction `plot()`.

Index

`barplot()`, 2

`hist()`, 2

`jpeg()`, 4

`par()`, 3

`pdf()`, 4

`plot()`, 3

boîtes et moustaches, 2

couleur, 2

diagramme à bâtons, 2

dispersion, 3

histogramme, 2

légende, 2

lignes, 3

paramètres graphiques, 3

plusieurs graphiques par page, 3

sauvegarde, 3

types de fichier, 4