



TD 7 à 10 Graphiques dans R

ING1 EC503 Algorithmique & programmation

Galharret Jean-Michel
département MSC

https://galharret.github.io/WEBSITE/cours_ONIRIS.html

Ouvrez la base de données iris

```
data("iris")
```

On peut faire un résumé de ce jeu de données au moyen de la fonction `summary()`

```
summary(iris)
```

Cas univarié :

On veut uniquement représenter une variable.

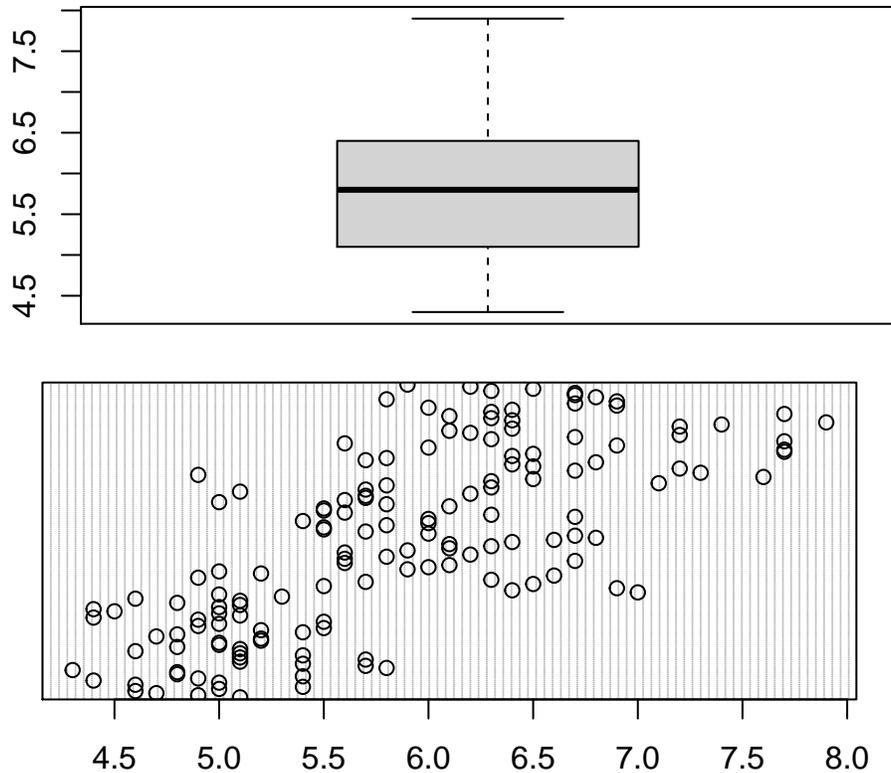
Graphiques pour une variable quantitative

On utilise principalement les trois types de graphiques suivants :

- boxplot (boîte à moustache),
- dotchart (points par points),

- `hist` (histogramme).

On va s'intéresser uniquement à la longueur des Sépales et réaliser les trois types de graphique :



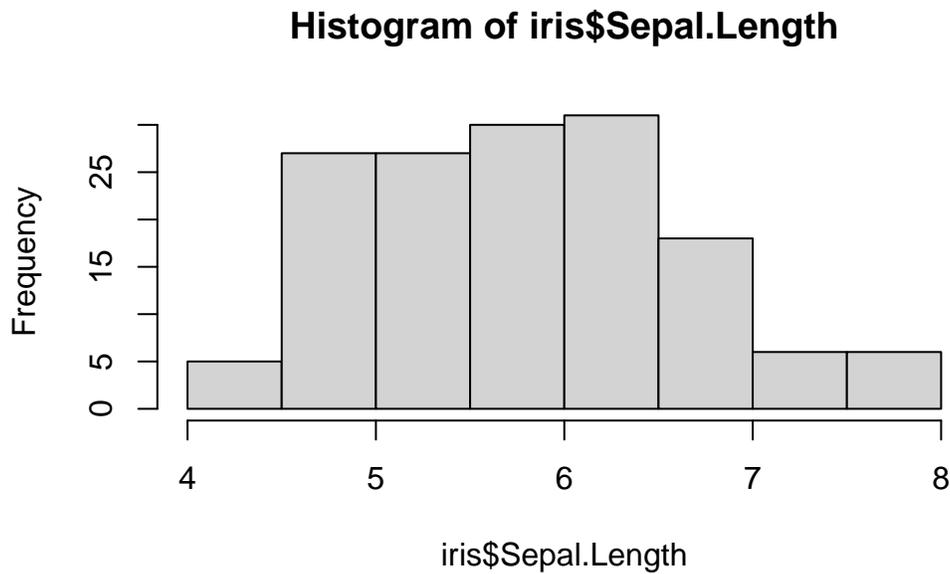
Les graphiques dans R offrent de très nombreux paramètres :

- titre principal *main*,
- titre des axes *xlab*, *ylab*
- couleur des graphiques *col*
- type des points *pch*

```
boxplot(iris$Sepal.Length,main="Longueur des sépales",ylab="L(cm)",col="purple")
dotchart(iris$Sepal.Length,main="Longueur des sépales",
         xlab="L(cm)",col="purple",pch=15)
```

On peut également superposer plusieurs graphiques grâce à la fonction *lines()* qui permet de dessiner des lignes. Cette fonction devra toujours être précédée d'un graphique.

On peut utiliser les arguments *lwd* (épaisseur du trait), *lty* (type de ligne) c'est à dire continue, pointillés, ...



```
hist(iris$Sepal.Length,main="Longueur des s pales",
     xlab="L(cm)",col="purple",prob=T)
lines(density(iris$Sepal.Length),lwd=3,col=2,lty=2)
```

On peut  galement utiliser la fonction *abline* pour ajouter une droite sur un graphique existant. Les param tres possibles sont :

- *h*= lorsque la droite est horizontale,
- *v*= lorsque la droite est verticale,
- *a* =, *b*= lorsque la droite est oblique du type $y = a + bx$ (attention *a* est l'ordonn e   l'origine et *b* le coefficient directeur).

```
hist(iris$Sepal.Length,main="Longueur des s pales",xlab="L(cm)",col="purple",prob=T)
lines(density(iris$Sepal.Length),lwd=3,col=2,lty=2)
abline(v=mean(iris$Sepal.Length),col=2,lwd=2,lty=2)
```

Graphiques pour une variable qualitative.

On utilise principalement les deux types de graphiques suivants :

- barplot (diagramme en b tons),
- pie (camembert ou diagramme circulaire).

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

Cas bivarié

On veut représenter une variable en fonction d'une autre.

Une variable quantitative en fonction d'une qualitative

On veut représenter la longueur des sépales en fonction de l'espèce sur un boxplot :

```
boxplot(iris$Sepal.Length~iris$Species)
```

ou bien en points par points (on va utiliser la couleur pour distinguer les espèces) :

```
dotchart(iris$Sepal.Length,col=iris$Species)
legend("bottomright",col=1:3,pch=rep(1,3),legend = levels(iris$Species),cex=0.5)
```

Remarque : Pour la position de la légende c'est une association top ou bottom avec left ou right. Il faut également préciser le type de points avec pch et on peut gérer la taille de la légende avec cex.

On peut aussi faire plusieurs histogrammes. On va scinder la fenêtre graphique en plusieurs parties *par(mfrow=c(v,h))* où v est le nombre de séparations verticales et h horizontales. Ici on a trois espèces d'iris donc on prend i=3 et j=1 :

```
par(mfrow=c(1,3))
hist(iris$Sepal.Length[iris$Species==levels(iris$Species)[1]],
     main=paste("Pour l'espèce ",levels(iris$Species)[1]),
     xlab="Longueur du Sépale",ylab="Fréquence")
hist(iris$Sepal.Length[iris$Species==levels(iris$Species)[2]],
     main=paste("Pour l'espèce ",levels(iris$Species)[2]),
     xlab="Longueur du Sépale",ylab="Fréquence")
hist(iris$Sepal.Length[iris$Species==levels(iris$Species)[3]],
     main=paste("Pour l'espèce ",levels(iris$Species)[3]),
     xlab="Longueur du Sépale",ylab="Fréquence")
```

Une variable quantitative en fonction d'une autre variable quantitative

C'est la fonction plot qui permet de faire ce type de représentation.

On représente la longueur des sépales en fonction de leur largeur (width). Comme précédemment on peut colorer les points en fonction de l'espèce et ajouter une légende.

```
plot(iris$Sepal.Width,iris$Sepal.Length,col=iris$Species,pch=20,
     xlab="largeur",ylab="longueur")
legend("topleft",legend=levels(iris$Species),col=1:3,pch=rep(20,3),cex=0.5)
```

Une variable qualitative en fonction d'une autre variable qualitative

On va créer trois classes d'iris (colonne nommée class) en fonction de la longueur du sépale : petit (inférieur à 5), moyen (entre 5 et 6) et grand (au delà de 6). On utilise la fonction cut de R :

```
iris$class<-cut(iris$Sepal.Length,breaks = c(0,5,6,10))
iris$class<-factor(iris$class,labels=c("petit","moyen","grand"))
```

On représente alors la table de contingence de la classe définie ci-dessus en fonction de l'espèce.

```
mosaicplot(table(iris$Species,iris$class),
            color = 2:4,
            main="répartition des classes en fonction des espèces")
```

Exercices

Exercice 1 :

Télécharger le fichier `temperat1.csv` sur CONECT puis ouvrez le dans RStudio (utiliser l'interface graphique), la première colonne correspondra aux noms des lignes du data.frame.

Premier travail : Reproduire la Figure 1.

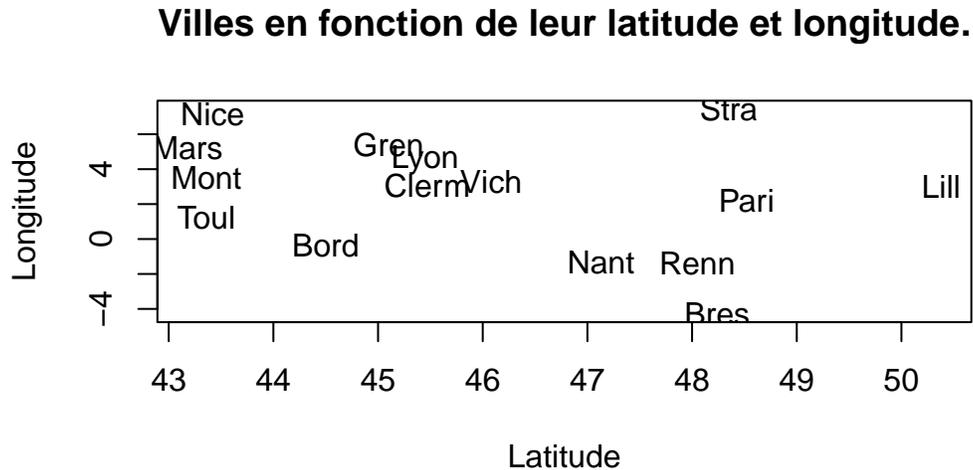


Figure 1: Représentation des villes

aide : pour ne pas avoir de points sur le graphe on utilise `type="n"` et on utilise la fonction `text()` [voir l'aide de R].

Deuxième travail : Représenter les températures moyennes pour les trois villes : Nantes, Brest et Rennes (Figure 2).

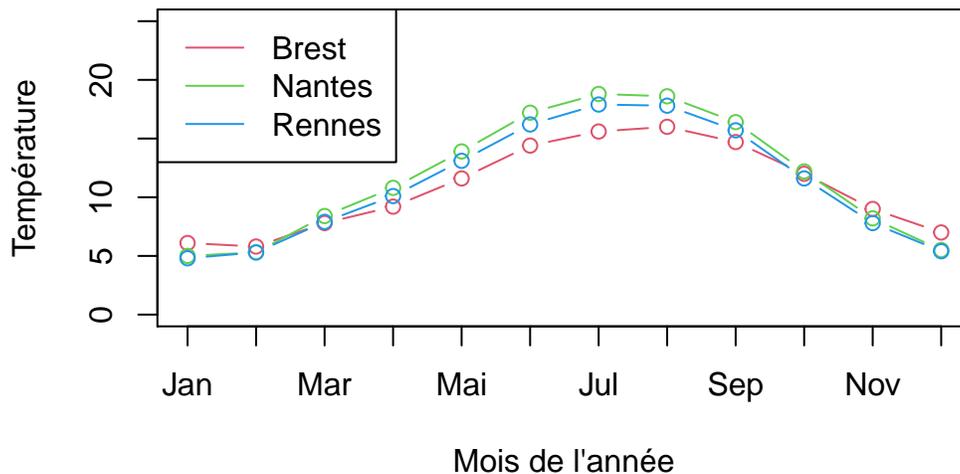


Figure 2: Températures moyennes mensuelles pour Brest, Nantes et Rennes

Pour mettre les mois en labels sur l'axe des abscisses on utilisera l'argument `xaxt="n"` dans `plot` et ensuite on utilisera la fonction `axis()` voir l'aide de R. L'argument `type="b"` pourra être utilisé dans `plot` et dans `lines` pour dessiner les points et les courbes.

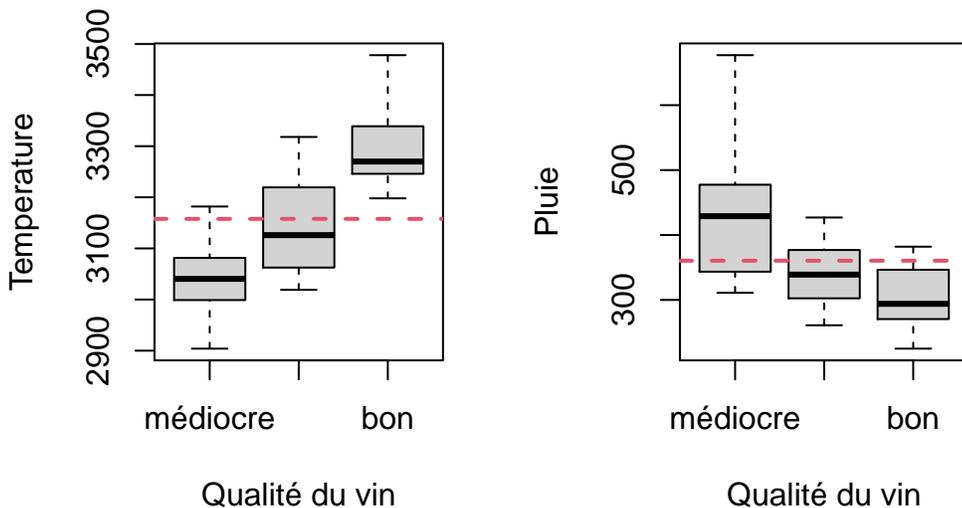
Exercice 2 :

Télécharger sur Connect le fichier `bordeaux1.csv` et ouvrez le dans RStudio à l'aide de l'interface graphique. Les années seront utilisées pour nommer les lignes.

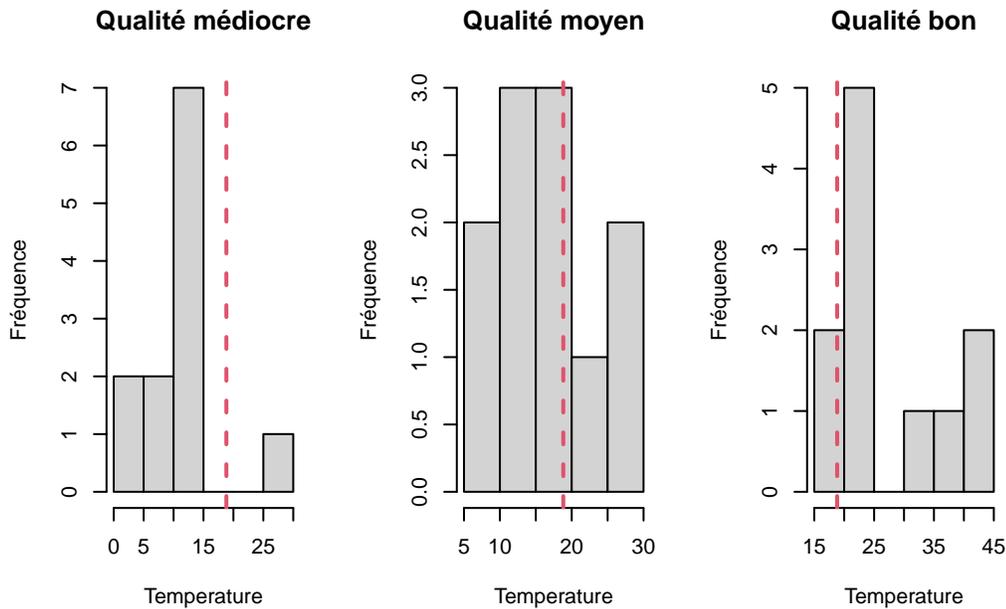
Il s'agit des données des vins de Bordeaux issues de "Méthodes statistiques en gestion" (M. Tenenhaus, Dunod, 1994).

On cherche à étudier l'éventuelle relation entre la qualité du vin et les variables météorologiques. Les variables sont les suivantes : `TEMPERATURE` : somme des températures moyennes journalières, `SOLEIL` : durée d'insolation, `CHALEUR` : nombre de jours de grande chaleur, `PLUIE` : hauteur des pluies, `Y (QUALITE DU VIN)` : 1 (Bon), 2 (Moyen), 3 (Médiocre).

1. Faire les boxplot des variables température et pluie en fonction de la qualité du vin. Les deux graphes seront dans une même fenêtre. La ligne en pointillé horizontale représente la moyenne globale de la variable (voir fonction `abline()`). On reclassera les modalités de la variable qualité.



2. Faire les 3 histogrammes de la variable chaleur en fonction de la qualité du vin dans une même fenêtre graphique. La droite verticale correspond à la moyenne globale.

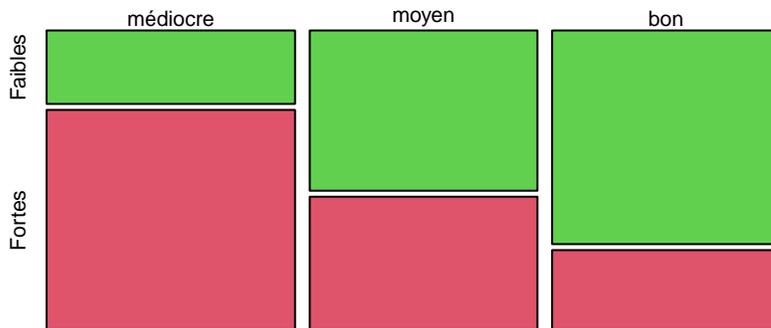


3. Créer une variable `pluie_quali` qui correspondra à la variable `PLUIE` et dont les modalités seront :

- Faibles lorsque la variable `PLUIE` sera inférieure à sa médiane
- Fortes lorsque la variable `PLUIE` sera supérieure ou égale à sa médiane.

Dessiner la répartition de la qualité des vins en fonctions des deux modalités Faibles et Fortes.

Répartition de la qualité en fonction de la quantité de plu



4. Reproduire le graphique suivant

Données climatiques associées aux vins

