TP 1
Prise en main de R
Statistiques exploratoire

L'objectif de ce TP est de se familiariser avec le logiciel scientifique R et un environnement de développement intégré RStudio.

1 Installation

Le logiciel R est disponible sur une grande variété de plateformes. Pour l'installer sur votre ordinateur télécharger l'archive correspondant à votre système à partir de l'adresse *https://pbil.univ-lyon1.fr/CRAN/*. Nous utiliserons l'environnement de développement intégré multiplateforme RStudio qui est disponible à l'adresse *https://www.rstudio.com/products/rstudio/download/#download*.

La fenêtre de démarrage du logiciel RStudio se compose principalement d'une console. Il s'agit d'une zone qui accepte les instructions en R, les exécute immédiatement en tapant sur le bouton *Entrée* et affiche le résultat lorsqu'il y en a un. Les instructions tapées dans cette zone ne sont pas sauvegardées. Pour éviter de retaper entièrement une instruction erronée ou entrée précédemment, il existe un historique des commandes déjà entrées. Pour naviguer dans cet historique, il suffit d'utiliser les touches *flèche haute* et *flèche basse*. Pour sauvegarder vos commandes R, il faut créer un fichier de *script*. Pour exécuter ces instructions, on pourra au choix exécuter la totalité du script en cliquant sur l'icône *Source* ou exécuter la ligne courante ou la sélection courante avec l'icône *Run.*

Pour accéder à la documentation d'une fonction, par exemple *sum()*, il suffit d'exécuter l'instruction *help(sum)* ou *?sum*.

2 Premiers pas

R est un logiciel de calcul scientifique très puissant, mais il fait aussi officie de calculatrice avec les opérateur arithmétique +, -, *, /, ** ainsi que les fonctions usuelles :

```
> 8*7
> 2**12
> 1+3+5+7+9
> 8*5/2
```

```
> sqrt(49)
> cos(pi/4)**2
> log10(1000)
> log(exp(1.6))
```

Q1 Vérifiez à l'aide de la fonction *identical()* que

$$\pi \approx \frac{\ln(640320^3 + 744)}{\sqrt{163}}$$

Pour plus de clarté dans les scripts, il est utile de stocker les résultats intermédiaires. La syntaxe en R utilise l'opérateur d'affectation <- pour stocker un résultat d'une variable.

```
> x <-  (3+4)/ sqrt(5)
```

La variable *a* contient désormais le nombre $\frac{3+4}{\sqrt{5}}$ qu'on pourra réutiliser en cas de besoin :

```
> sqrt(a)
```

```
> a**5
```

3 Structure de données usuelles

Le langage R met à disposition des structures de données adaptées à la nature des données statistiques usuelles. Nous allons voir les trois plus simples : le vecteur, le facteur et le tableau individus-variables.

Vecteur : le vecteur est la structure de données la plus simple en R. Elle regroupe une collection de données de même type. L'instruction suivante définit ainsi un vecteur des quatre éléments 2, 3, 5, 0 dans cet ordre :

```
> c(2, 3, 5, 0)
```

Q2 Créer le vecteur *notes* contenant les notes suivantes :

8, 11.5, 19.5, 5.5, 5, 5.5, 10.5, 4.5, 0.5

On peut également concaténer des vecteurs. L'instruction suivante donne un résultat identique à *c(2, 3, 5, 0).*

```
> c(2, c(3, 5, 0))
```

Q3 Rajouter la note 14 au vecteur *notes*

Une des forces de R est d'avoir une syntaxe simple permettant d'appliquer la même opération sur chaque élément d'un ou de plusieurs vecteurs. Tester les instructions suivantes :

```
> v <- c(0, 1, 2, 3, 4)
> u <- c(3, 5, 7, 9, 9)
> v + u
> 5 * v
```

```
> u / 3
> u * v
> u ==3
> v > pi
```

Q4 Ramenez les notes sur 10. Combien d'étudiants ont eu la moyenne ?

Indice : TRUE et FALSE sont aussi considérés, respectivement, comme 1 et 0.

Pour extraire un sous-vecteur d'un vecteur, on utilise la notation [ ] en spécifiant les indices qui nous intéressent :

```
> v[1]
> v[2]+v[4]
```

```
> v[2 :4]
> v[c(1, 3, 5)]
```

On remarquera que les indices des cases d'un vecteur commence à 1 et non pas à 0 comme c'est le cas dans beaucoup d'autres langages.

Q5 Quelle est la moyenne de la première, cinquième et dernière note ?

Indice : la fonction *length()* peut être utile.

Un autre moyen d'extraction très utile est de spécifier ce qu'on appelle un masque. Il s'agit d'un tableau de Booléen de même taille que le vecteur, qui indique si on souhaite garder un élément ou pas.

```
> Choix <- v>pi
> v[choix]
```

Q6 Combien d'étudiant ont eu la moyenne ? (utilisez cette fois l'extraction et la fonction *length()*).

Q7 Quelle est la plus petite note entière ?

Indice : utilisez les deux fonctions *floor()* et *min()*.

Il est possible de modifier les valeurs des éléments d'un vecteur en utilisant l'opérateur d'affectation :

```
> v[5] <- 0
> v[2 : 4] <- 1
> v[v==1] <- 2
```

Q8 On enlève 2 points à chaque étudiant. Créez le vecteur *notes2* des notes abaissées de 2 points, combien y a-t-il des notes négatives ? Mettez-les à zéro.

Q9 A quoi peuvent bien servir les fonctions suivantes :

rev(), *sort()*, *prod()*, *mean()*, *range()*, *median()* et *str()*.

Testez ces foncions sur le vecteur *notes*.

Facteur : les vecteurs modélisent des échantillons d'une variable quantitative. Pour manipuler efficacement des échantillons d'une variable qualitative, on utilise une variante du vecteur qu'on appelle le facteur. Pour créer un facteur à partir d'une collection de données on utilise la fonction *factor()*.

```
> (couleur <- c("J", "R", "B", "R"))    # ce sont les couleur primaire Jaune Rouge et Bleu
[1] "J" "R" "B"
> (f <- factor(couleur))
[1] J R B R
Levels : B J R
```

Remarquons que le croisillon « # » permet d'écrire les commentaires et les parenthèses autour d'une expression force l'affichage du résultat.

On reconnait un facteur au fait qu'il spécifie les modalités de la variable qualitative. La syntaxe pour l'extraction et la modification des facteurs reste la même. Néanmoins, certaines opérations n'ont pas de sens quand la variable qualitative n'est pas ordonnée. Si on désire fixer un ordre entre les modalités de la variable qualitative on spécifie *ordered=TRUE* en argument supplémentaire de *factor()*. En revanche, l'ordre retenu est l'ordre alphabétique, pour fixer un ordre différent il faut ajouter l'argument *levels* lors de l'appel de la fonction *factor()*.

```
> (f <- factor(couleur, ordered = TRUE, levels = c("J", "R", "B"))
[1] J R B R
Levels : J < R < B
```

On cherche à modéliser la séquence ADN suivante :

$$ACAAGATGCCATTGTC$$

Q10 Créez le facteur *adn.seq* modélisant la séquence ADN. Testez les fonctions *levels() et nlevels()*.

Q11 Cette séquence contient combien de nucléotide de chaque type ?

Indice : il suffit de trouver la taille des sous-facteurs qui ne contiennent que les nucléotides d'un même type.

Tableau individus-variables : la structure de donnée qui modélise un tableau individus-variables est un *data.frame*. Pour les créer, il faut spécifier les colonnes en arguments. Une colonne est représentée par un vecteur si c'est une variable quantitative et par un facteur quand c'est une variable qualitative.

On dispose de 6 sacs, dans chaque sac on trouve un nombre de ballon de même couleur :

```
> nombre <- c(5, 9, 1, 7, 13, 7)
> couleur <- factor( c("J", "R","J", "R", "B", "R"))
> X <- data.frame(nombre, couleur)
> X
```

La dernière instruction nous permet de visualiser le résultat de ce script :

	nombre	Couleur
1	5	J
2	9	R
3	1	J
4	7	R
5	13	B
6	7	R

Lors de l'affichage, R numérote les individus de la population lorsque leurs noms ne sont pas spécifiés et donne des noms aux colonnes lorsqu'ils sont absents.

En pratique, un *data.frame* peut être stockés dans des fichiers au format CSV. Pour les utiliser il faut les charger en mémoire avec la fonction *read.csv()* en spécifiant le chemin absolu.

Chargez le jeu de donnée stocké dans le fichier LCS.data fourni avec le TP avec l'instruction suivante :

```
> Y <- read.csv("LCS.data")
```

Q12 Que font les fonctions *length()*, *ncol()*, *nrow()*, *names()*, *head()* et *summary()*.

Pour extraire des données d'un tableau individus-variables, on utilise une syntaxe similaire à celle des vecteurs et des facteurs à une différence près qu'il y a désormais deux dimensions donc deux indices à spécifier, séparés par une virgule.

```
> X[3,2]          # extraire l'élément qui se trouve à la première colonne et la première ligne
> X[,2]                                                  # extraire la deuxième colonne
> X[4,]                                                  # extraire la quatrième ligne
> X[c(3,5,1),]          # extraire la troisième, la cinquième et la première ligne dans cet ordre
```

Q13 Qui a corrigé le CC ainsi que l'examen de l'étudiant 53 et quelle est la note de CC et de l'examen ainsi que le mention de l'étudiant 167 ?

On peut se servir du nom d'une colonne directement avec la syntaxe *Y$CC* qui est équivalent à *Y[,1]* et on peut le manipuler directement comme un vecteur et écrire par exemple *Y$CC[2]* pour afficher la note de CC de l'étudiant 2.

Q14 Combien d'étudiant ont eu la moyenne à l'examen ?

Quelle est la moyenne des CC corrigés par le correcteur EG ?

Quelle est la proportion des étudiants qui ont progressé ?

Quel est le correcteur d'examen le plus dur dans sa correction ?

4 Statistiques descriptives

Dans cette section, nous allons voir comment extraire des informations pertinente de ces données.

Q15 Quelle sont les statistiques classiques sur les notes d'examen ? et comparez avec la fonction *summary()*.

Indice : utilisez les fonctions *mean()*, *sd()*, *var()*, *median()*, *min()*, *max()*, *quantile()* et *IQR()*.

Q16 Calculer la moyenne tronquée d'ordre 10 ?

Indice : n'oubliez pas d'ordonner l'échantillon avant de tronquer

5 Analyse univariée

Dans cette section, nous allons passer en revue les différentes fonctions que R met à notre disposition pour visualiser les données en fonction de leur nature.

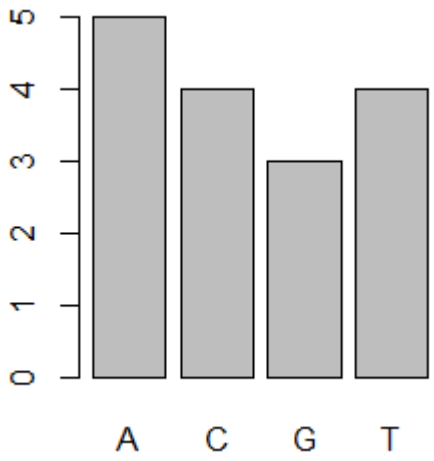
5.1 Variables qualitatives

Diagramme en bâtons : le diagramme en bâtons permets d'afficher le nombre d'occurrences de chaque modalité dans le jeu de données considérées. Pour tracer un diagramme en bâtons, on va faire appel à deux fonction : *table()* et *barplot()*.

La première fonction renvoie une table de contingence qui compte le nombre d'occurrence de chaque modalité. En reprenant l'exemple de la séquence ADN, on a :

```
> (t <- table(adn.seq))
adn.seq
A C G T
5 4 3 4
```

Il suffit ensuite d'appeler la deuxième fonction avec cette table : *barplot(t)*.



Q17 Faire le diagramme en bâtons du nombre de copies corrigées par correcteur. Quel est le correcteur qui a corrigé le plus de copie ?

5.2 Variables quantitatives

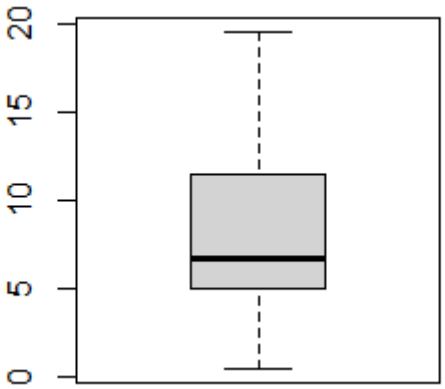
Histogramme : l'histogramme permet de représenter la répartition d'un jeu de données dans un ensemble d'intervalles formant une partition. Lorsque les intervalles sont tous de même longueur, R choisit de représenter en ordonnées les effectifs. Dans le cas contraire, R calcul les ordonnées de manières à ce que l'air de chaque bande de l'histogramme soit proportionnelle à l'effectif. Le coefficient de proportionnalité est fixé tel que l'air total soit égale à 1.

Q18 Utiliser la fonction *hist()* pour tracer l'histogramme des notes d'examen.

Q19 On voudrait séparer les étudiants ayant eu plus de 15 des étudiant ayant eu moins.

Indice : utiliser le paramètre *breaks* de la fonction *hist()* pour spécifier un tel découpage.

Boite à moustaches : c'est un résumé numérique d'une variable quantitative qui permet d'indiquer les statistiques renvoyées par la fonction *summary()*. En R, on utilise la fonction *boxplot()* sur le vecteur des données. Sur l'exemple du vecteur *notes* on obtient :



Q20 Tracer la boite à moustaches des notes d'examens. Trouver le nombre de valeurs aberrantes.

Indice : les valeurs aberrantes sont les valeurs en dehors de la boite à moustaches.

Indice 2 : la moustache inférieure est la donnée immédiatement supérieure au premier quartile moins 1.5 fois l'étendue interquartiles.

6 Analyse bivariée

6.1 Quantitatives vs quantitatives

Lorsque les variables sont quantitatives, chaque individu est caractérisé par deux nombres réels correspondant à un point dans le plan. On trace ces points en faisant appel à la fonction *plot()* avec les deux vecteurs en arguments séparés par un ~.

Q21 Tracer le graphique de dispersion des notes de l'examen par rapport au CC.

6.2 Quantitatives vs qualitatives

Q22 Tracer les boites à moustaches des notes de l'examen en fonction des correcteurs.

Lorsque l'échantillon est de taille raisonnable on peut faire un graphique de dispersion unidimensionnel à la place des boites à moustaches. On utilise alors la fonction *stripchart()*.

Q23 Tracer le graphe des dispersions des notes de l'examen en fonction des correcteurs.

Q24 Que fait l'argument *method="jitter"* lorsqu'il est spécifié dans *stripchart()*.