Statistiques Descriptives avec R

Christophe Ambroise

2025-03-12

Table des matières

Préface

Ce livret propose des exercices et leurs solutions pour le cours ‘’Statistiques descriptives avec R’’.

1 Langage R

Exercice 1.1. Types

Dans cet exercice, nous allons utiliser les trois types de variables disponibles dans R :

1. Créer deux variables booléennes (TRUE, FALSE) et tester toutes les opérations logiques.
2. Utiliser R comme une calculatrice pour calculer et stocker dans une variable la moyenne des notes suivantes :
$$10, 11, 15, 16, 21$$
3. Manipuler une chaîne de caractères : transformer la chaîne “Statistiques descriptives avec R” en séparant tous les mots.

Solution 1.1.

```
# 1. Opérations logiques avec des booléens
a <- TRUE
b <- FALSE
a | b  # OU logique
```

```
[1] TRUE
```

```
a & b  # ET logique
```

```
[1] FALSE
```

```
# 2. Calcul de la moyenne
notes <- c(10, 11, 15, 16, 21)
moyenne <- mean(notes)
moyenne  # Affichage du résultat
```

```
[1] 14.6
```

```r
# 3. Manipulation de texte
phrase <- "Statistiques descriptives avec R"
mots <- strsplit(phrase, " ")
mots  # Affichage des mots séparés
```

```
[[1]]
[1] "Statistiques" "descriptives" "avec"         "R"
```

Exercice 1.2. Génération de Vecteurs

Dans cet exercice, nous allons nous familiariser avec la création de vecteurs en R en utilisant les commandes c(), seq(), rep(), paste() et leurs options.

1. Créer un vecteur contenant la suite des entiers de 1 à 12 de deux manières différentes.
2. Créer le vecteur

```
c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0)
```

de trois manières différentes.

3. Créer un vecteur contenant tous les multiples de 2 compris entre 1 et 50.
4. Créer un vecteur contenant tous les nombres de 1 à 100 qui ne sont pas des multiples de 5.
5. Créer un vecteur contenant 3 fois chacun des 10 chiffres.
6. Créer un vecteur contenant une fois la lettre A, deux fois la lettre B,..., 26 fois la lettre Z. Quelle est la longueur de cette suite ? (Utiliser la chaîne LETTERS prédéfinie).
7. Créer le vecteur suivant :

```
c("individu 1", "individu 2", ..., "individu 100")
```

Solution 1.2. **Génération de Vecteurs**

```r
# 1. Suite des entiers de 1 à 12 de deux manières différentes
v1 <- c(1:12)
v2 <- seq(1, 12, by=1)

# 2. Vecteur avec valeurs décimales de trois manières différentes
v3_1 <- c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0)
v3_2 <- seq(0.5, 5.0, by=0.5)
v3_3 <- (0:9) / 2 + 0.5

# 3. Tous les multiples de 2 entre 1 et 50
```

```r
v4 <- seq(2, 50, by=2)

# 4. Tous les nombres de 1 à 100 qui ne sont pas des multiples de 5
v5 <- (1:100)[(1:100) %% 5 != 0]

# 5. 3 fois chacun des 10 chiffres
v6 <- rep(0:9, each=3)

# 6. Une fois A, deux fois B, ..., 26 fois Z
v7 <- rep(LETTERS, times=1:26)
longueur_v7 <- length(v7)  # Longueur de la séquence

# 7. Vecteur avec "individu 1" à "individu 100"
v8 <- paste("individu", 1:100)

# Affichage des résultats
list(v1, v2, v3_1, v3_2, v3_3, v4, v5, v6, v7, longueur_v7, v8)
```

Exercice 1.3. **Comptage**

Ecire un code R qui génère un échantillon $(x_1, \cdots, x_100)$ de loi parente $\mathcal{N}(0, 2)$:

1. compter le nombre de réalisations supérieures à 1.
2. donner l'effectif théorique de réalisations supérieures à 1.

Solution 1.3. **Comptage**

Version vectorielle

```r
set.seed(3)
sum(rnorm(100,mean=0,sd=2) > 1)
```

```
[1] 32
```

```r
(1-pnorm(1,sd=2))*100
```

```
[1] 30.85375
```

D'un point de vue effectif théorique, nous nous attendons à avoir $100.P(X > 1)$: 30.8537539

Version boucle

```r
set.seed(3)
x<-rnorm(100,mean=0,sd=2)
compteur<-0
for (i in 1:100)
  if (x[i]>1) compteur<-compteur+1
print(compteur)
```

```
[1] 32
```

Exercice 1.4. **Résumé de Texte**

Écrire une série d'instructions R qui résume l'extrait suivant de Wikipedia :

> La statistique est d'un point de vue théorique une science, une méthode et une technique. La statistique comprend : la collecte des données, le traitement des données collectées, l'interprétation des données, la présentation afin de rendre les données compréhensibles par tous. Remarquons que la statistique est parfois notée « la Statistique » (avec une majuscule) ce qui permet de différencier cette science avec une statistique (avec une minuscule). Le pluriel est également souvent utilisé pour désigner ce domaine mathématique : « les statistiques », cela permet de montrer la diversité de cette science ; le pluriel est utilisé en anglais. Ainsi la statistique est un domaine des mathématiques qui possède une composante théorique ainsi qu'une composante appliquée. La composante théorique est proche de la théorie des probabilités et forme avec cette dernière, les sciences de l'aléatoire. La statistique appliquée est utilisée dans presque tous les domaines de l'activité humaine : ingénierie, management, économie, biologie, informatique, etc. La statistique utilise des règles et des méthodes sur la collecte des données, pour que celles-ci puissent être correctement interprétées, souvent comme composante d'une aide à la décision. Le statisticien a pour profession la mise au point d'outils statistiques, dans le secteur privé ou le secteur public, et leur exploitation généralement dans un domaine d'expertise.

Une approche rapide consiste à prendre la première et la dernière phrase de cet extrait.

Solution 1.4.

```r
options(width = 60)
# Définition du texte initial
MonTexte <- "La statistique est d'un point de vue théorique une science,
une méthode et une technique. La statistique comprend : la collecte des données,
le traitement des données collectées, l'interprétation des données, la présentation
```

```
afin de rendre les données compréhensibles par tous.
Remarquons que la statistique est parfois notée
« la Statistique » (avec une majuscule) ce qui permet
de différencier cette science avec une statistique
(avec une minuscule). Le pluriel est également souvent
utilisé pour désigner ce domaine mathématique :
« les statistiques », cela permet de montrer la diversité de cette science ;
le pluriel est utilisé en anglais.
Ainsi la statistique est un domaine des mathématiques qui possède
une composante théorique ainsi qu'une composante appliquée.
La composante théorique est proche
de la théorie des probabilités et forme avec cette dernière,
les sciences de l'aléatoire. La statistique
appliquée est utilisée dans presque tous les domaines de l'activité humaine :
ingénierie, management,
économie, biologie, informatique, etc.
La statistique utilise des règles et des méthodes sur la collecte
des données, pour que celles-ci puissent être correctement interprétées,
souvent comme composante
d'une aide à la décision. Le statisticien a pour profession
la mise au point d'outils statistiques, dans le
secteur privé ou le secteur public, et leur exploitation
généralement dans un domaine d'expertise."

# Fonction pour extraire le résumé
resume <- function(MonTexte) {
  phrases <- unlist(strsplit(MonTexte, split = "\\. "))
  cat(paste(phrases[1], " ...", phrases[length(phrases)], sep = ""))
}

# Résumé du texte
resume(MonTexte)
```

```
La statistique est d'un point de vue théorique une science,
une méthode et une technique ...Le statisticien a pour profession
la mise au point d'outils statistiques, dans le
secteur privé ou le secteur public, et leur exploitation
généralement dans un domaine d'expertise.
```

Exercice 1.5. Parcourir une liste d'achats

Écrire un code R qui parcourt une liste d'articles (par exemple : pain, carottes, Nutella) décrits chacun par :

- leur prix (respectivement : 2, 1, 8 euros),
- leur nutriscore (respectivement : B, A, D).

Le programme devra :

1. Afficher le prix de chaque article,
2. Calculer et afficher le total des achats.

Solution 1.5.

```
# Définition de la liste d'articles
articles <- list(
  list(nom = "pain", prix = 2, nutriscore = "B"),
  list(nom = "carottes", prix = 1, nutriscore = "A"),
  list(nom = "nutella", prix = 8, nutriscore = "D")
)
# Affichage des prix et calcul du total
total <- 0
for (article in articles) {
  cat(article$nom, ":", article$prix, "euros\n")
  total <- total + article$prix
}
```

```
pain : 2 euros
carottes : 1 euros
nutella : 8 euros
```

```
cat("Total des achats :", total, "euros\n")
```

```
Total des achats : 11 euros
```

2 Outils de rédaction et d'analyse reproductible

Exercice 2.1. Ecire un document minimal qui permette de produire un fichier html, word, beamer, pdf, powerpoint.

Solution 2.1.

```
---
title: "Exercice série 1"
output:
  pdf_document:
    toc: true
  beamer_presentation: default
  html_document:
    toc: true
    toc_float: true
  word_document: default
  powerpoint_presentation: default
date: "2025-03-14"
---
```

Exercice 2.2. **Rédaction d'un document Quarto pour résoudre une équation quadratique**

1. Commencez le document avec une introduction expliquant l'objectif de résoudre l'équation quadratique suivante :

$$x^2 - 5x + 6 = 0$$

2. Résolvez l'équation en détaillant les étapes mathématiques.
3. Vérifiez la solution numérique de l'équation en utilisant R ou Python.
4. Enregistrez le document sous le format HTML et PDF.

Solution 2.2.

Listing 2.1 Code yaml

```yaml
---
title: "Résolution d'une équation quadratique"
author: "Étudiant"
date: today
format:
  html: default
  pdf: default
lang: fr
---
```

Listing 2.2 Code markdown

```
# Introduction

Dans ce document, nous allons résoudre l'équation quadratique suivante :

$$ x^2 - 5x + 6 = 0 $$

Nous utiliserons la formule du discriminant et vérifierons les solutions en Python.

# Résolution mathématique

L'équation quadratique standard est donnée par :

$$ ax^2 + bx + c = 0 $$

Ici, nous avons \( a = 1 \), \( b = -5 \) et \( c = 6 \).

Le discriminant est :

$$ \Delta = b^2 - 4ac = (-5)^2 - 4(1)(6) = 25 - 24 = 1 $$

Les solutions sont données par :

$$ x = \frac{-b \pm \sqrt{\Delta}}{2a} $$

En remplaçant les valeurs :

$$ x_1 = \frac{5 + 1}{2} = \frac{6}{2} = 3 $$
$$ x_2 = \frac{5 - 1}{2} = \frac{4}{2} = 2 $$

Les solutions sont **\( x = 2 \) et \( x = 3 \)**.

# Vérification en Python
Nous pouvons confirmer ces solutions en utilisant Python :
```

Listing 2.3 Code python

````
```python
#| eval: true
#| include: true
# Définition des coefficients de l'équation ax^2 + bx + c = 0
a = 1
b = -5
c = 6

# Calcul du discriminant Δ = b^2 - 4ac
delta = b * b - 4 * a * c

# Résolution en fonction de la valeur de Δ
if delta > 0:
    x1 = (-b + (delta ** 0.5)) / (2 * a)  # √Δ est remplacé par ** 0.5
    x2 = (-b - (delta ** 0.5)) / (2 * a)
    solutions = [x1, x2]
elif delta == 0:
    x1 = -b / (2 * a)  # Cas où Δ = 0, une seule solution
    solutions = [x1]
else:
    solutions = ["Pas de solution réelle"]

# Affichage des solutions
print(solutions)
```
````

Listing 2.4 Code R

```r
```

```r
#| eval: true
#| include: true

# Définition des coefficients de l'équation ax^2 + bx + c = 0
a <- 1
b <- -5
c <- 6

# Calcul du discriminant Δ = b^2 - 4ac
delta <- b * b - 4 * a * c

# Résolution en fonction de la valeur de Δ
if (delta > 0) {
  x1 <- (-b + sqrt(delta)) / (2 * a)  # √Δ est calculé avec sqrt()
  x2 <- (-b - sqrt(delta)) / (2 * a)
  solutions <- c(x1, x2)
} else if (delta == 0) {
  x1 <- -b / (2 * a)  # Cas où Δ = 0, une seule solution
  solutions <- c(x1)
} else {
  solutions <- "Pas de solution réelle"
}

# Affichage des solutions
print(solutions)
```

```
```

3 Statistiques avec R: les bases

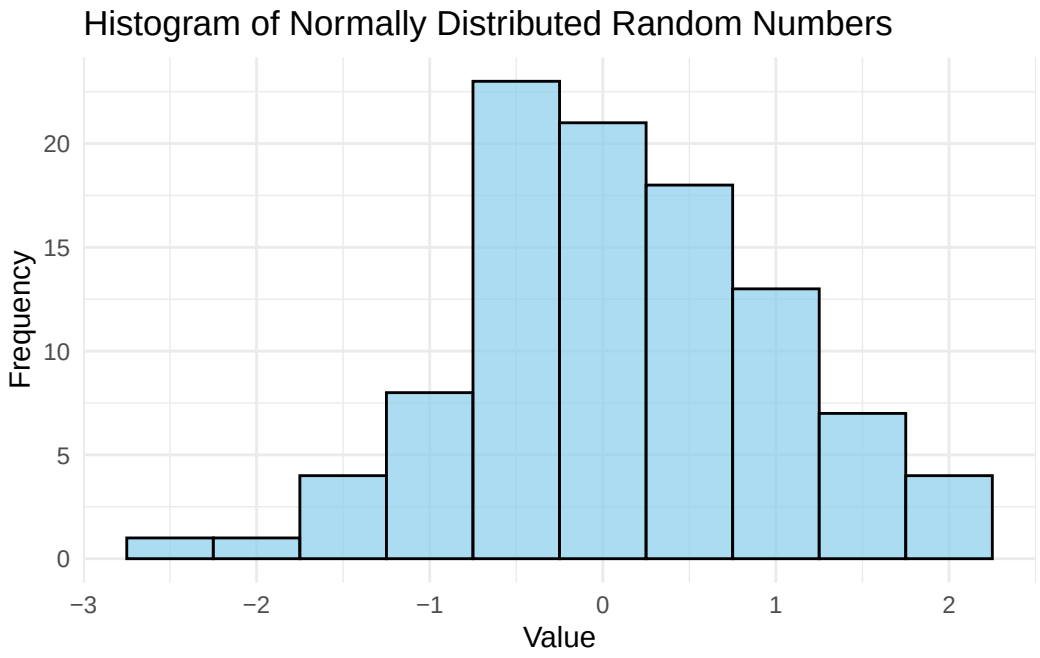
Exercice 3.1. **Utilisation d'un type de distribution**

Illustrer avec la loi normale, les quatre fonctions R:

- `rnorm` simulation
- `dnorm` densité
- `pnorm` fonction de répartition
- `qnorm`quantile

Solution 3.1. **Simulation et histogramme**

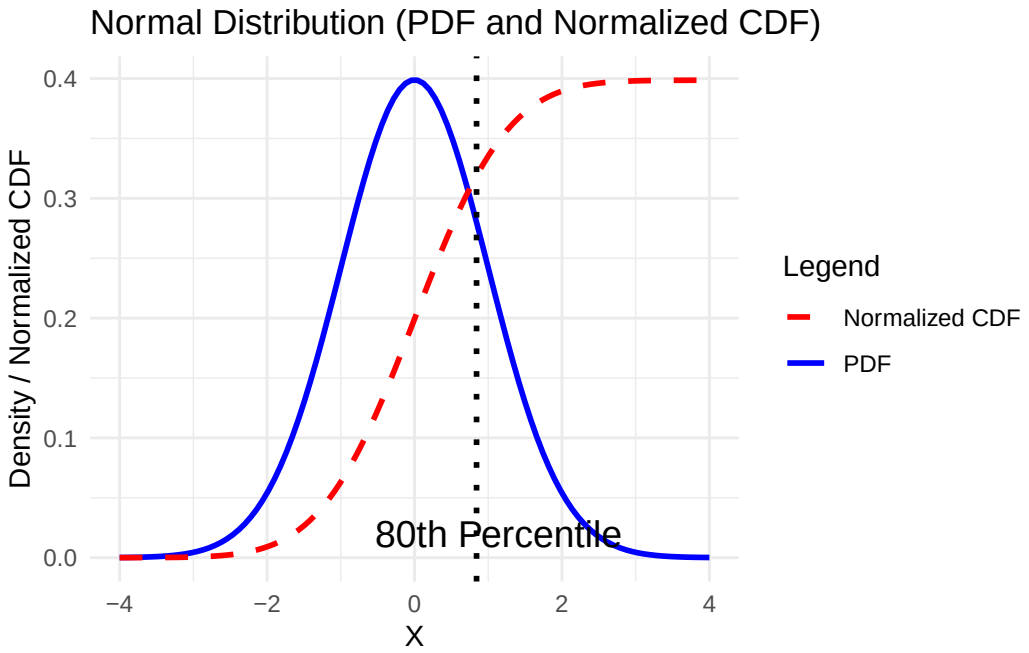
```r
library(ggplot2)

# Set seed for reproducibility
set.seed(123)

# Parameters for the normal distribution
mean <- 0
sd <- 1

# Generate a sample
x <- rnorm(100, mean = mean, sd = sd)

# Create a data frame for ggplot2
data <- data.frame(x = x)

# Plot histogram using ggplot2
ggplot(data, aes(x = x)) +
  geom_histogram(binwidth = 0.5, fill = "skyblue", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Normally Distributed Random Numbers",
       x = "Value",
       y = "Frequency") +
  theme_minimal()
```



Densité, fonction de répartition et quantile

```r
# Generate x values
x_vals <- seq(-4, 4, length.out = 100)

# Compute PDF and CDF values
pdf_values <- dnorm(x_vals, mean = mean, sd = sd)
cdf_values <- pnorm(x_vals, mean = mean, sd = sd)

# Normalize CDF values to be on a similar scale as PDF
cdf_values_normalized <- cdf_values / max(cdf_values) * max(pdf_values)

# Compute the 80th percentile quantile
quantile_80 <- qnorm(0.8, mean = mean, sd = sd)

# Create a data frame for ggplot2
data <- data.frame(
  x = x_vals,
  pdf = pdf_values,
  cdf_normalized = cdf_values_normalized
)

# Plot using ggplot2
ggplot(data, aes(x = x)) +
```

```r
  geom_line(aes(y = pdf, color = "PDF"), linewidth = 1) +
  geom_line(aes(y = cdf_normalized, color = "Normalized CDF"), linetype = "dashed", linewidth
  geom_vline(xintercept = quantile_80, linetype = "dotted", color = "black", linewidth = 1) +
  annotate("text", x = quantile_80 + 0.3, y = 0.02, label = "80th Percentile", color = "black
  labs(title = "Normal Distribution (PDF and Normalized CDF)",
       x = "X",
       y = "Density / Normalized CDF") +
  scale_color_manual(name = "Legend", values = c("PDF" = "blue", "Normalized CDF" = "red")) +
  theme_minimal()
```



Exercice 3.2. **Loi normale et écart à la moyenne**

Soit $X \sim \mathcal{N}(\mu, \sigma^2)$. Calculer la probabilité que

$$P(X \in [\mu - k\sigma, \mu + k\sigma]),$$

pour $k \in \{1, 2, 3, 4\}$.

Solution 3.2.

$$P(X \in [\mu - k\sigma, \mu + k\sigma]) = P(X \leq \mu + k\sigma) - P(X \leq \mu - k\sigma) \tag{3.1}$$
$$= P\left(\frac{X-\mu}{\sigma} \leq k\right) - P\left(\frac{X-\mu}{\sigma} \leq -k\right) \tag{3.2}$$
$$= \phi(k) - \phi(-k) \tag{3.3}$$
$$= 2\phi(k) - 1 \tag{3.4}$$

```r
library(knitr)
tableau=data.frame(k=1:4,Pk=2*pnorm(1:4)-1)
tableau_transpose <- as.data.frame(t(tableau))
colnames(tableau_transpose) <- paste0("k=", tableau$k)
tableau_transpose <- cbind(Variable = rownames(tableau_transpose), tableau_transpose)
rownames(tableau_transpose) <- NULL

# Affichage avec kable
kable(tableau_transpose, caption = "Tableau transposé de k et Pk")
```

Table 3.1: Tableau transposé de k et Pk

Variable	k=1	k=2	k=3	k=4
k	1.0000000	2.0000000	3.0000000	4.0000000
Pk	0.6826895	0.9544997	0.9973002	0.9999367

Exercice 3.3. **Estimation du poids moyen des nouveau-nés**

Une maternité a enregistré le poids de **100 bébés** nés à terme. Ces poids sont modélisés par une variable aléatoire suivant une **loi normale** de moyenne (= 3.3) kg et d'écart-type (= 0.5) kg.

Dans le cadre de cette étude, trois valeurs **atypiques** (outliers) ont été malencontreusement ajoutées à l'échantillon : **2000 kg, 3000 kg et 3500 kg**.

Objectifs

- Estimer la **moyenne empirique** et l'**écart-type empirique** de l'échantillon **avec** et **sans** les valeurs aberrantes.
- Visualiser l'effet des outliers sur les estimations et sur l'histogramme des poids.

Simulation des données

```
set.seed(42)
poids <- rnorm(100, mean = 3.3, sd = 0.5)
poids_outliers <- c(poids, 2000, 3000, 3500)
```

Solution 3.3. **Fonctions robustes pour moyenne et écart-type tronqués**

```
# Fonction pour la moyenne tronquée (trimmed mean)
mean.trimmed <- function(x, alpha = 0.2) {
  borne.inf <- quantile(x, probs = alpha / 2)
  borne.sup <- quantile(x, probs = 1 - alpha / 2)
  normal <- (x > borne.inf) & (x < borne.sup)  # Sélection des données non extrêmes
  x.trimmed <- x[normal]                       # Échantillon tronqué
  return(mean(x.trimmed))
}

# Fonction pour l'écart-type tronqué
sd.trimmed <- function(x, alpha = 0.2) {
  borne.inf <- quantile(x, probs = alpha / 2)
  borne.sup <- quantile(x, probs = 1 - alpha / 2)
  normal <- (x > borne.inf) & (x < borne.sup)
  x.trimmed <- x[normal]
  return(sd(x.trimmed))
}
```

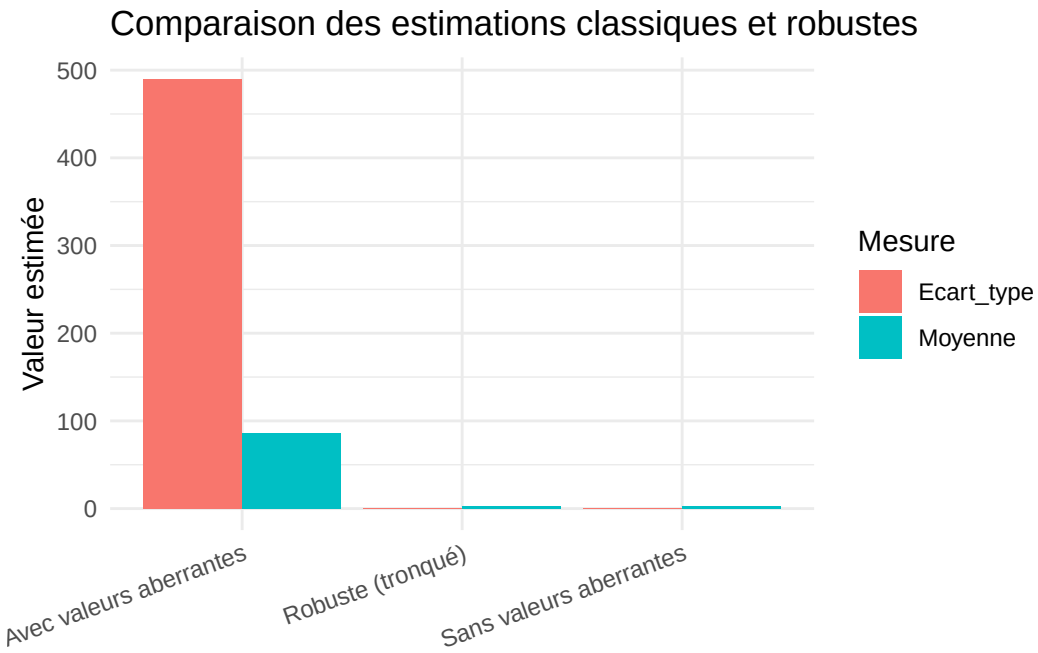
Estimation des moyennes et écarts-types

```
# Moyenne et écart-type classiques (sans valeurs aberrantes)
mean_sans <- mean(poids)
sd_sans <- sd(poids)

# Moyenne et écart-type classiques (avec valeurs aberrantes)
mean_avec <- mean(poids_outliers)
sd_avec <- sd(poids_outliers)

# Moyenne et écart-type robustes (avec valeurs aberrantes)
mean_trimmed <- mean.trimmed(poids_outliers)
sd_trimmed <- sd.trimmed(poids_outliers)

# Tableau récapitulatif
moyennes <- data.frame(
```

```
  Cas = c("Sans valeurs aberrantes", "Avec valeurs aberrantes", "Robuste (tronqué)"),
  Moyenne = c(mean_sans, mean_avec, mean_trimmed),
  Ecart_type = c(sd_sans, sd_avec, sd_trimmed)
)

# Affichage tableau ggplot
library(ggplot2)
library(tidyr)
library(dplyr)
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
donnees_long <- pivot_longer(moyennes, cols = c(Moyenne, Ecart_type), names_to = "Mesure", va

ggplot(donnees_long, aes(x = Cas, y = Valeur, fill = Mesure)) +
  geom_col(position = "dodge") +
  labs(title = "Comparaison des estimations classiques et robustes",
       y = "Valeur estimée", x = NULL) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 20, hjust = 1))
```



Analyse

- L'ajout des **valeurs aberrantes** modifie beaucoup la moyenne et l'écart-type classiques sur notre expérience.
- Les estimations **robustes** (moyenne et écart-type tronqués) sont **moins sensibles** aux valeurs extrêmes.
- Cela illustre l'intérêt de méthodes robustes en présence de données bruitées ou atypiques.
- Ces techniques sont utiles en pratique pour fournir une **meilleure estimation centrale** lorsque l'on suspecte la présence de perturbations dans les données.

Exercice 3.4. **Illustration du Théorème central limite (TCL) à l'aide de simulations de lancers de pièce***

On considère une expérience aléatoire consistant à lancer une pièce de monnaie équilibrée (pile ou face) un nombre fixe de fois `q`, et à compter le nombre de fois où l'on obtient **pile**. Cette expérience est répétée `n` fois.

1. Quel est le nom de la loi de probabilité qui régit le nombre de piles obtenus lors d'une série de `q` lancers ?
2. Quelle est sa moyenne et sa variance ?
3. Simulez cette expérience `n` fois, et représentez sous forme d'histogramme la distribution du nombre de piles obtenus.
4. Superposez à cet histogramme la densité d'une loi normale de même moyenne et écart-type.
5. Que peut-on conclure ? Quel théorème cela illustre-t-il ?

Solution 3.4. **Loi suivie par le nombre de piles**

Lorsqu'on lance une pièce `q` fois et que l'on compte le nombre de piles, ce nombre suit une **loi binomiale** de paramètres `q` (nombre d'essais) et `p = 0.5` (probabilité d'obtenir pile à chaque lancer).
Ainsi, si on note `X` le nombre de piles, alors :

$$X \sim \mathcal{B}(q, p)$$

- **Moyenne et variance**

Pour une variable aléatoire `X` suivant une loi binomiale $\mathcal{B}(q, p)$, la moyenne μ et la variance σ^2 sont données par :

- **Espérance** :

$$\mathbb{E}[X] = q \cdot p$$

- **Variance** :

$$\mathrm{Var}(X) = q \cdot p \cdot (1 - p)$$

Dans notre cas, avec `p = 0.5`, on a :

$$\mathbb{E}[X] = \frac{q}{2}, \quad \mathrm{Var}(X) = \frac{q}{4}$$

Simulation et visualisation

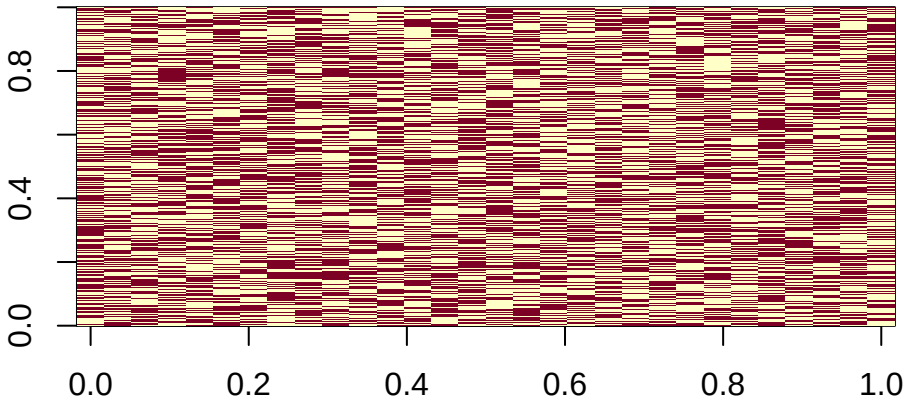
On simule `n` expériences indépendantes, chacune consistant à lancer une pièce `q` fois et à compter le nombre de piles obtenus. Les résultats sont stockés dans un vecteur, puis représentés sous forme d'histogramme.

L'objectif est d'observer que la **distribution empirique** du nombre de piles se rapproche visuellement d'une **loi normale**, ce qui illustre le **théorème central limite**.

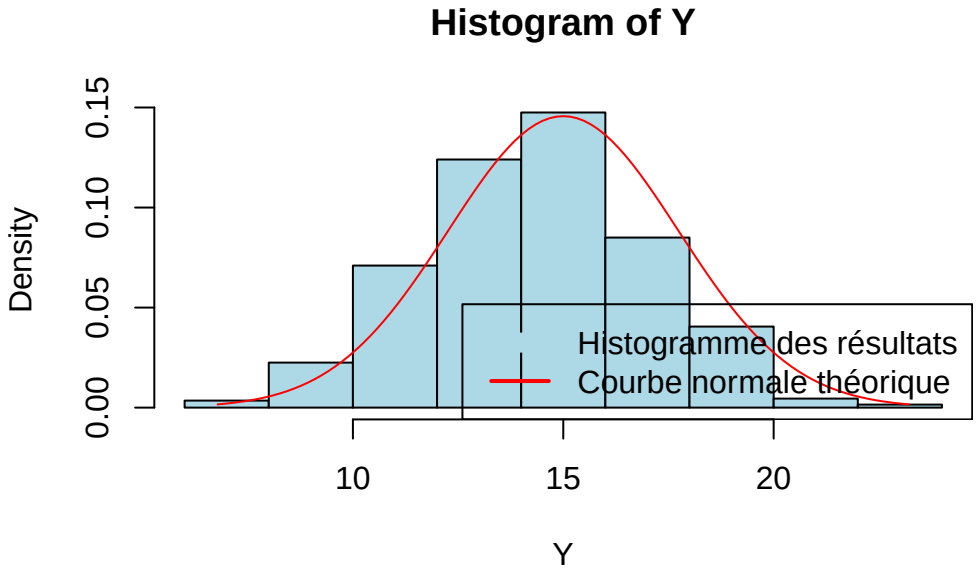
```r
# Une expérience (q lancers)
q<-30
sample(0:1,q,replace=TRUE)
```

```
 [1] 1 1 0 1 0 0 1 1 1 1 1 0 1 0 1 0 1 1 0 1 0 0 0 0 1 1 1 1 0 0
```

```r
# n expériences dans une matrice X ou chaque ligne est une expérience
n<-1000
X<-matrix(sample(0:1,n*q,replace=TRUE),nrow=n,ncol = q, byrow = TRUE)
image(t(X))
```




```r
Y<-rowSums(X)
hist(Y,probability = TRUE, col = "lightblue")
curve(dnorm(x,mean=0.5*q,sd=sqrt(q*0.5^2)),0.5*q -  3* sqrt(q*0.5^2), 0.5*q +  3* sqrt(q*0.5
# Ajouter une légende
legend("bottomright", legend = c("Histogramme des résultats", "Courbe normale théorique"),
       col = c("lightblue", "red"), lwd = c(10, 2))
```



Exercice 3.5. **Simulation de tailles selon le sexe**

Simuler un échantillon d'individus en attribuant à chacun un sexe aléatoire ("H" pour homme, "F" pour femme), puis générer leur taille selon une loi normale dont les paramètres dépendent du sexe.

1. Créez un vecteur `sexe` de longueur `n = 100`, composé aléatoirement des lettres `"H"` (homme) et `"F"` (femme), avec une probabilité égale pour chaque sexe.
2. Créez un vecteur `taille` de même longueur destiné à contenir les tailles des individus.
3. Pour chaque individu, simulez une taille suivant une loi normale :
   - Si l'individu est un homme (`"H"`), la taille est simulée selon une loi normale de moyenne **178 cm** et d'écart-type **5 cm**.
   - Si l'individu est une femme (`"F"`), la taille est simulée selon une loi normale de moyenne **167 cm** et d'écart-type **5 cm**.
4. Affichez les 10 premières valeurs des vecteurs `sexe` et `taille` pour vérification.
5. Calculez et affichez :
   - la moyenne et l'écart-type des tailles **pour l'ensemble de l'échantillon**,
   - la moyenne et l'écart-type **séparément pour les hommes et les femmes**.
6. Représentez graphiquement la distribution des tailles à l'aide d'un **histogramme**, en utilisant des couleurs différentes pour les hommes et les femmes.

- Cet exercice permet de manipuler des vecteurs, des instructions conditionnelles, la génération de nombres aléatoires selon une loi normale, et la visualisation de données en R.
- Il peut être réalisé à l'aide de boucles (`for`) ou avec une approche vectorisée.

Solution 3.5. Voici deux solutions. La première utilise très peu de fonctions R et la seconde utilise R de manière plus optimale.

Solution 1

```r
n <- 100
taille <- rep(0, n)
sexe <- rep("", n)

# Génération des sexes et des tailles
for (i in 1:n) {
  sexe[i] <- sample(c("H", "F"), 1)
  if (sexe[i] == "H") {
    taille[i] <- rnorm(1, mean = 178, sd = 5)
  } else {
```

```
    taille[i] <- rnorm(1, mean = 167, sd = 5)
  }
}

# Initialisation des vecteurs de moyennes et effectifs
moyennes <- c("F" = 0, "H" = 0)
effectifs <- c("F" = 0, "H" = 0)

# Accumulation des tailles et des effectifs
for (i in 1:n) {
  if (sexe[i]=="F") {
    effectifs["F"]<-effectifs["F"]+1
    moyennes["F"]<-moyennes["F"]+taille[i]
   } else

   {
    effectifs["H"]<-effectifs["H"]+1
    moyennes["H"]<-moyennes["H"]+taille[i]
   }

 }

# Calcul des moyennes finales
moyennes <- moyennes / effectifs
```

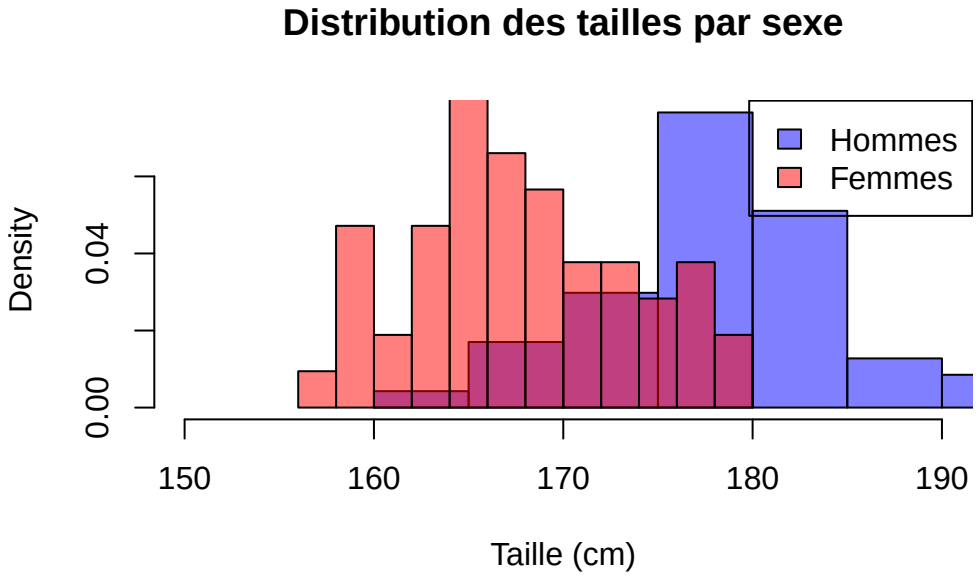
Solution 2

```
n <- 100
sexe <- sample(c("H", "F"), n, replace = TRUE)
taille <- ifelse(sexe == "F", rnorm(n, 167, 5), rnorm(n, 178, 5))
df <- data.frame(sexe, taille)

# Moyenne de taille par sexe
tapply(df$taille, df$sexe, mean)
```

```
       F        H
167.7108 177.9976
```

```
# Histogramme avec couleurs par sexe
hist(taille[sexe == "H"], col = rgb(0, 0, 1, 0.5), xlim = c(150, 190),
     main = "Distribution des tailles par sexe", xlab = "Taille (cm)",
```

```r
     breaks = 10, freq = FALSE)
hist(taille[sexe == "F"], col = rgb(1, 0, 0, 0.5), add = TRUE, breaks = 10, freq = FALSE)

legend("topright", legend = c("Hommes", "Femmes"),
       fill = c(rgb(0, 0, 1, 0.5), rgb(1, 0, 0, 0.5)))
```



Exercice 3.6. **Manipulation de matrices : prix de voitures en euros et dollars**

Travailler la manipulation de matrices en R en simulant des données économiques liées à la vente de voitures en Europe et aux États-Unis.

1. Créez une matrice `X` à 3 lignes et 2 colonnes contenant :
   - en première colonne, le **prix de fabrication** de 3 voitures,
   - en seconde colonne, le **prix de vente en France** pour ces 3 modèles.
2. Nommez les colonnes respectivement `"Fabrication"` et `"Vente.France"`.
3. Calculez une nouvelle matrice `X_US` correspondant aux mêmes prix **exprimés en dollars américains**, en utilisant un taux de conversion euro/dollar égal à **1,08**.
4. À partir de `X_US`, calculez une nouvelle matrice `Y` contenant 3 colonnes :
   - le **prix de fabrication** (en dollars),
   - le **prix de vente en France** (en dollars),
   - le **prix de vente aux États-Unis**, obtenu en appliquant une **taxe de 25%** sur le prix de vente en France (en dollars).

5. Affichez la matrice finale `Y` ainsi que la moyenne de chaque colonne.

Solution 3.6.

```r
# Création de la matrice initiale en euros
X <- matrix(c(10000, 33000,
              4500, 15000,
              50000, 5000), nrow = 3, ncol = 2, byrow = TRUE)
colnames(X) <- c("Fabrication", "Vente.France")

# Affichage de la matrice X
print(X)
```

```
     Fabrication Vente.France
[1,]       10000        33000
[2,]        4500        15000
[3,]       50000         5000
```

```r
# Taux de conversion euro → dollar
tauxEU.US <- 1.08

# Conversion de tous les montants en dollars
X_US <- X * tauxEU.US

# Matrice de transformation pour ajouter la colonne Vente.US (avec 25% de taxe)
A <- matrix(c(1, 0, 0,
              1, 0, 1.25), nrow = 2, ncol = 3, byrow = TRUE)

# Création de la matrice finale Y
Y <- X_US %*% A
colnames(Y) <- c("Fabrication", "Vente.France", "Vente.US")

# Affichage de la matrice finale
print(Y)
```

```
     Fabrication Vente.France Vente.US
[1,]       46440            0    44550
[2,]       21060            0    20250
[3,]       59400            0     6750
```

```r
# Moyenne des colonnes
colMeans(Y)
```

```
 Fabrication Vente.France     Vente.US
       42300            0        23850
```

Exercice 3.7. **Centrage-réduction (Z-transformation) : données iris**

La **centrage-réduction**, ou **Z-transformation**, transforme chaque variable en une nouvelle variable de **moyenne 0** et d'**écart-type 1**.

Pour chaque valeur x, on calcule :

$$z = \frac{x - \mu}{\sigma}$$

où :

- μ est la moyenne de la variable (colonne),
- σ est l'écart-type.

Travailler le centrage-réduction de données en R sur les 4 premières variables du jeu de données iris.

1. Chargez les données iris et extrayez les colonnes 1 à 4 dans une matrice X.
2. Calculez la moyenne et l'écart-type de chaque colonne à l'aide de la fonction `apply()`.
3. Centrez et réduisez manuellement les colonnes de X pour obtenir une matrice Z dans laquelle :

- chaque valeur est soustraite de la moyenne de sa colonne,
- puis divisée par l'écart-type de sa colonne.

4. Vérifiez que la moyenne de chaque colonne de Z est proche de 0 et que l'écart-type est proche de 1.
5. Affichez un histogramme pour chaque colonne de la matrice standardisée.
6. Ecrire le centrage réduction comme une opération d'algèbre linéaire et donner une interprétation géométrique.

Solution 3.7.

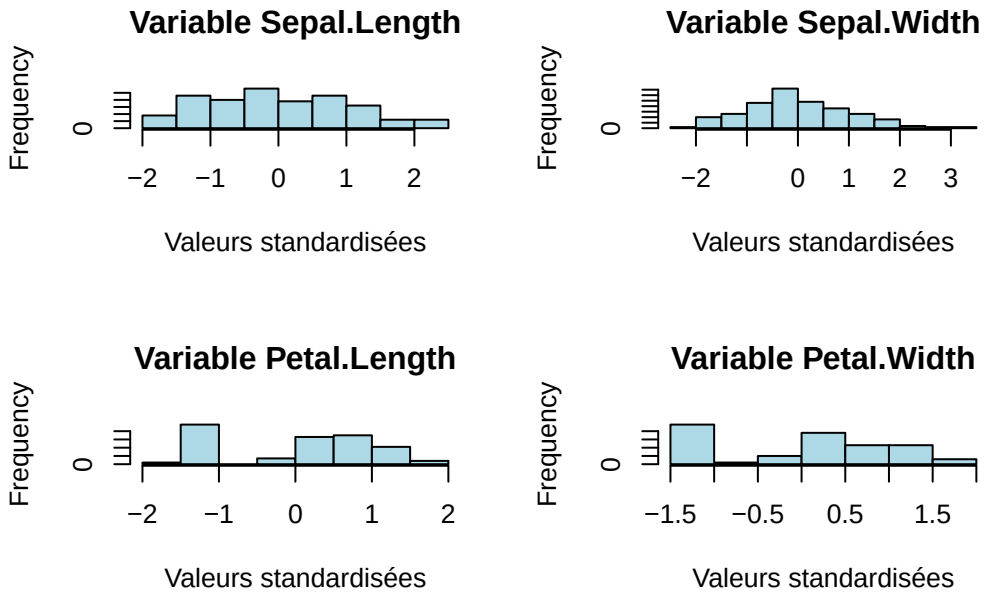
```r
# Étape 1 : Charger les données et extraire les colonnes numériques
data(iris)
X <- as.matrix(iris[, 1:4])

# Étape 2 : Moyenne et écart-type par colonne avec apply()
moy <- apply(X, 2, mean)
ecart <- apply(X, 2, sd)

# Étape 3 : Centrage-réduction "à la main" avec apply()
Z <- apply(X, 2, function(col) (col - mean(col)) / sd(col))

# Étape 4 : Vérification des moyennes et écarts-types
apply(Z, 2, mean)  # Doit être proche de 0
```

```
 Sepal.Length   Sepal.Width  Petal.Length   Petal.Width
-1.554405e-15  7.096175e-16  1.361874e-16 -2.886580e-16
```

```r
apply(Z, 2, sd)    # Doit être proche de 1
```

```
Sepal.Length  Sepal.Width Petal.Length  Petal.Width
           1            1            1            1
```

```r
# Étape 5 : Histogrammes des colonnes de Z
par(mfrow = c(2, 2))
for (i in 1:4) {
  hist(Z[, i],
       main = paste("Variable", colnames(X)[i]),
       xlab = "Valeurs standardisées",
       col = "lightblue")
}
```




```
# Étape 6 : Algèbre linéaire
X<-as.matrix(X)
X.bar<-cbind(rep(1,150))%*%rbind(apply(X,2,mean))
S.inv<-diag(apply(X,2,function(x){1/sd(x)}))
Z<-(X-X.bar)%*%S.inv
```

6.

Soit une matrice de données $X \in \mathbb{R}^{n \times p}$ avec n observations (lignes) et p variables (colonnes).

On veut obtenir une matrice centrée-réduite $Z \in \mathbb{R}^{n \times p}$, où chaque colonne a **moyenne nulle** et **écart-type unitaire**.

Écriture matricielle

1. **Centrage** :

On soustrait à chaque colonne de X sa moyenne, notée $\mu \in \mathbb{R}^p$:

$$X_c = X - \mathbf{1}_n \mu^\top$$

où : - $\mu = \frac{1}{n} X^\top \mathbf{1}_n$ - $\mathbf{1}_n$ est un vecteur colonne de taille n rempli de 1.

2. **Réduction** :

On divise chaque colonne j de X_c par son écart-type σ_j, ce qui revient à **multiplier** à droite par une matrice diagonale D^{-1}, avec :

$$D = \text{diag}(\sigma_1, ..., \sigma_p)$$

La matrice centrée-réduite s'écrit donc :

$$Z = (X - \mathbf{1}_n \mu^\top) D^{-1}$$

Interprétation géométrique

Centrer une variable revient à **translater** les données pour que le nuage de points soit centré sur l'origine de l'espace $\mathbb{R}^p$

Réduire une variable revient à **changer l'échelle** : chaque axe est transformé pour avoir une **longueur unitaire**, autrement dit on **normalise** l'espace.

Après centrage-réduction : - Le nuage de points est **centré à l'origine**, - Chaque variable a la même influence, **quelle que soit son unité ou son échelle**, - Les **distances euclidiennes** deviennent comparables entre observations ce qui est bien utile pour des méthodes géométriques comme :

- l'**ACP** (projection sur axes de variance maximale),
- le **Clustering** (k-means, etc.),
- la **Régression** avec pénalisation (ridge, lasso…).

Exercice 3.8. **Corrélation empirique et interprétation géométrique**

On considère deux échantillons centrés de taille n, notés :

$$x = (x_1, x_2, ..., x_n)^\top \in \mathbb{R}^n \quad \text{et} \quad y = (y_1, y_2, ..., y_n)^\top \in \mathbb{R}^n$$

c'est-à-dire que $\sum_{i=1}^n x_i = 0$ et $\sum_{i=1}^n y_i = 0$.

1. **(Norme et variance)**
   Montrez que la **norme euclidienne** de x est reliée à sa **variance empirique** $\text{Var}(x)$ par:

$$\|x\|^2 = \sum_{i=1}^n x_i^2 = n \cdot \text{Var}(x)$$

et de même pour y.

2. **(Corrélation et angle)**
   On rappelle que la **corrélation empirique** entre x et y est donnée par :

$$\mathrm{Corr}(x, y) = \frac{1}{n} \sum_{i=1}^{n} \frac{x_i}{\sigma_x} \cdot \frac{y_i}{\sigma_y}$$

où σ_x et σ_y sont les écarts-types de x et y.

Montrez que cette formule revient à calculer :

$$\mathrm{Corr}(x, y) = \cos(\theta)$$

où θ est l'**angle entre les vecteurs** x et y dans $\mathbb{R}^n$.

3. **(Interprétation)**
   Interprétez géométriquement les cas suivants :

   - Corrélation = 1
   - Corrélation = 0
   - Corrélation = –1

Solution 3.8 (Corrélation empirique et interprétation géométrique).

1. Norme et variance

Pour un échantillon centré $x = (x_1, x_2, \ldots, x_n)^\top \in \mathbb{R}^n$ avec $\sum_{i=1}^n x_i = 0$:

La variance empirique est :

$$\mathrm{Var}(x) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2 = \frac{1}{n} \sum_{i=1}^{n} x_i^2$$

car $\bar{x} = 0$ (échantillon centré).

En multipliant par n :

$$n \cdot \mathrm{Var}(x) = \sum_{i=1}^{n} x_i^2 = \|x\|^2$$

De même pour y.

2. Corrélation et angle

La corrélation empirique est :

$$\text{Corr}(x, y) = \frac{1}{n} \sum_{i=1}^{n} \frac{x_i}{\sigma_x} \cdot \frac{y_i}{\sigma_y}$$

Avec $\sigma_x = \sqrt{\frac{\|x\|^2}{n}} = \frac{\|x\|}{\sqrt{n}}$ et $\sigma_y = \frac{\|y\|}{\sqrt{n}}$:

$$\text{Corr}(x, y) = \frac{1}{n} \sum_{i=1}^{n} \frac{x_i}{\frac{\|x\|}{\sqrt{n}}} \cdot \frac{y_i}{\frac{\|y\|}{\sqrt{n}}} = \frac{1}{n} \cdot \frac{n}{\|x\| \cdot \|y\|} \cdot \sum_{i=1}^{n} x_i y_i = \frac{\langle x, y \rangle}{\|x\| \cdot \|y\|}$$

Or, $\cos(\theta) = \frac{\langle x,y \rangle}{\|x\| \cdot \|y\|}$ où θ est l'angle entre x et y.

Donc $\text{Corr}(x, y) = \cos(\theta)$.

3. Interprétation géométrique

- **Corrélation = 1** : $\cos(\theta) = 1 \implies \theta = 0°$. Vecteurs colinéaires de même sens. Relation linéaire parfaite positive.
- **Corrélation = 0** : $\cos(\theta) = 0 \implies \theta = 90°$. Vecteurs orthogonaux. Absence de relation linéaire.
- **Corrélation = -1** : $\cos(\theta) = -1 \implies \theta = 180°$. Vecteurs colinéaires de sens opposés. Relation linéaire parfaite négative.

```
gini<-function(x){
Num<-0
n<-length(x)
for(xi in x)
  for(xj in x)
     Num<-abs(xi-xj)+Num

return(G=Num/(n*(n-1)*2*mean(x)))}

gini(c(0,0,0,0,0,0,0,10))
```

```
[1] 1
```

```
gini(c(2,2,2,2,2))
```

```
[1] 0
```

```r
gini(c(2,4,5,6))
```

```
[1] 0.254902
```

Exercice 3.9. Soit l'échantillon $x = (1, 1, 1, 2, 1, 3, 1, 3, 1, 2, 2, 2)$

1. Quels sont les effectifs des modalités 1,2 et 3?
2. Comment les trouver avec une instruction R ?
3. Calculer la fréquence de chaque modalité
4. Calculer la fréquence cumulée ?
5. Ecrire une fonction qui calcule les 3 quantités à partir d'un échantillon x

Solution 3.9.

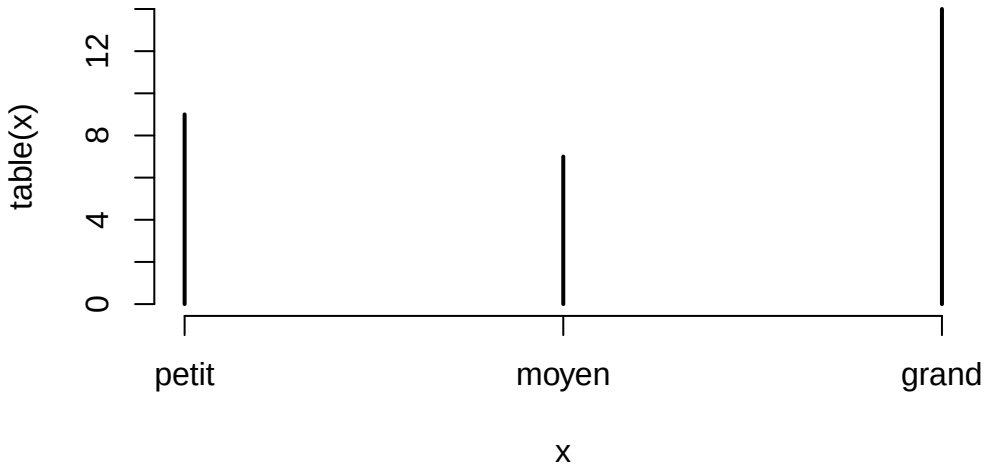
```r
x<-c(1,1,1,2,1,3,1,3,1,2,2,2)
effectifs<-table(x)
frequences<-table(x)/length(x)
frequences.cumulees<-cumsum(frequences)
cat("Les fréquences des modalités sont ", frequences, "\n", "Les fréquences cumulées sont ",
```

```
Les fréquences des modalités sont  0.5 0.3333333 0.1666667
 Les fréquences cumulées sont  0.5 0.8333333 1
```

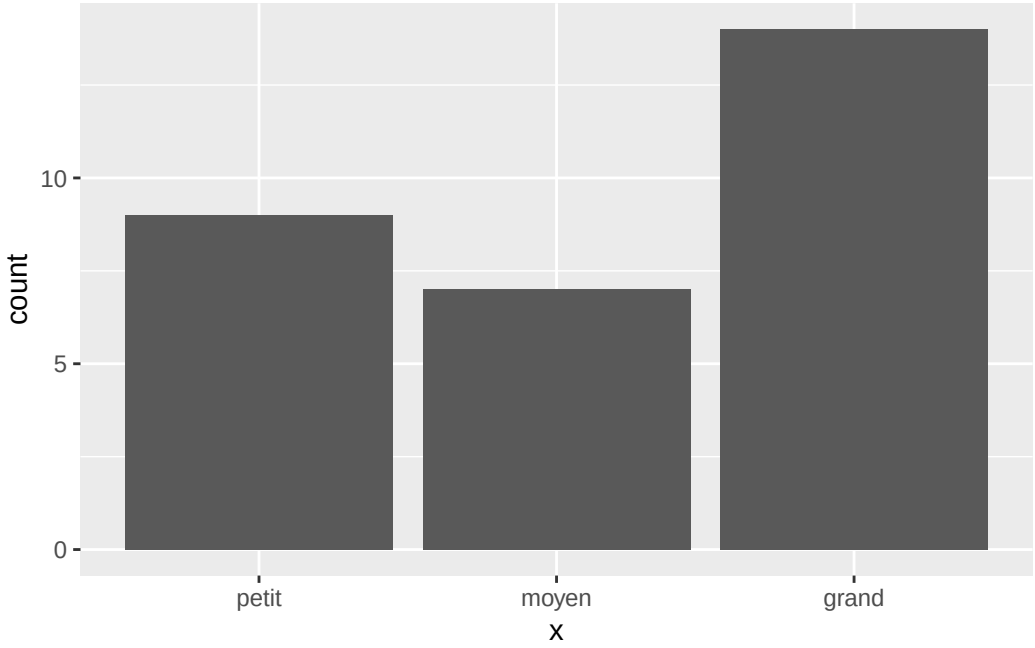
```r
stat.quali<-function(x){

effectifs<-table(x)
frequences<-table(x)/length(x)
frequences.cumulees<-cumsum(frequences)
return(list(effectifs=effectifs,frequences=frequences,frequences.cumulees=frequences.cumulees
}

resultat<-stat.quali(x)

# Cela fonctionne aussi avec les chaine de caractères
x<-sample(c("petit","moyen","grand"),30, replace=TRUE)
x<- factor(x, levels = c("petit", "moyen", "grand"), ordered = TRUE)
stat.quali(x)
```

```
$effectifs
x
petit moyen grand
    9     7    14

$frequences
x
    petit     moyen     grand
0.3000000 0.2333333 0.4666667

$frequences.cumulees
    petit     moyen     grand
0.3000000 0.5333333 1.0000000
```

```
plot(table(x))
```




```
library(ggplot2)
ggplot(data.frame(x),aes(x=x))+ geom_bar()
```



Exercice 3.10. Ecrire une fonction `fractile.emp` qui retourne le fractile empirique dr'odre α d'un échantillon x.

Solution 3.10.

```
fractile.emp<-function(x,alpha){
  n<-length(x)
  index <-ifelse(n*alpha == floor(n*alpha), n*alpha , floor(n*alpha)+1 )
  return(list(fractile=sort(x)[index],index=index))
}

x<-c(2,3,5,2000,1)
cat("x=",x,"\n")
```

```
x= 2 3 5 2000 1
```

```
cat("x trié=",sort(x),"\n")
```

```
x trié= 1 2 3 5 2000
```

```r
fractile<-fractile.emp(x,0.3)$fractile
cat("Le fractile d'ordre ", 0.3, " est ", fractile)
```

```
Le fractile d'ordre  0.3  est  2
```

```r
# En R

quantile(x,0.3)
```

```
30%
2.2
```