Rapport de projet statistique inférentielle en R

Table of contents

1 *Exercice 1 : Algortihme en R*

1. Calculer la moyenne et la variance d'une série statistique des données entrées par un utilisateur

- Première du code :
  - On demande à l'utilisateur de saisir l'éffectif de la série statiqtique
  - On initialise un vecteur 'numeric' de la longueur spécifiée
  - On demande de saisir les différents éléments de la série statistique

```r
#==========================================================================
#                          Exercice 1
#
#==========================================================================


# Demander à l'utilisateur la taille de la série
n <- as.integer(readline("Entrez l'effectif de la série : "))

# Initialiser un vecteur vide
serie <- numeric(n)

# Lire les éléments un par un
for (i in 1:n) {
  serie[i] <- as.numeric(readline(paste("Entrez la valeur numéro", i, ": ")))
}
```

2. Resultat de la première partie du code

```
> # Demander à l'utilisateur la taille de la série
> n <- as.integer(readline("Entrez l'effectif de la série : "))
Entrez l'effectif de la série : 5
> # Initialiser un vecteur vide
> serie <- numeric(n)
> # Lire les éléments un par un
> for (i in 1:n) {
+   serie[i] <- as.numeric(readline(paste("Entrez la valeur numéro", i, ": ")))
+ }
Entrez la valeur numéro 1 : 0.60
Entrez la valeur numéro 2 : 0.80
Entrez la valeur numéro 3 : 0.78
Entrez la valeur numéro 4 : 0.45
Entrez la valeur numéro 5 : 0.90
>
```

- Seconde partie du code :
  - On calcul la moyenne de la série puis on l'affiche
  - On calcul la variance de l'echantillon ()

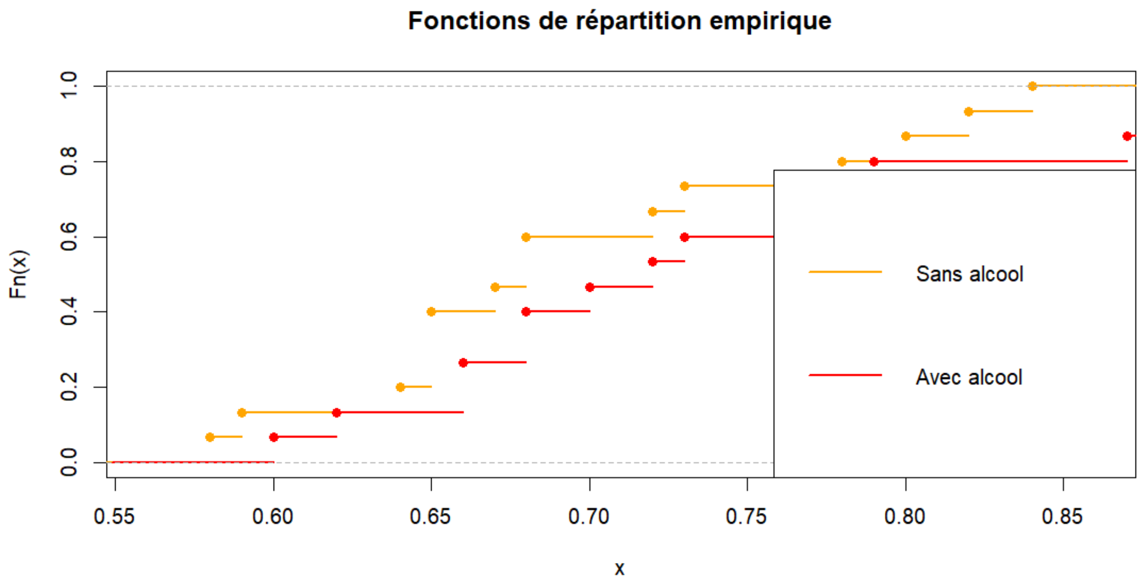
```r


# Calcul de la moyenne
moyenne <- mean(serie)

# Affichage du résultat
cat("La moyenne de la série est :", moyenne, "\n")

# Variance de l'échantillon
variance_echantillon = var(serie)
cat("La variance de l'échantillon est de: ", variance_echantillon, "\n")

# Calcul de la variance de la population (S²)
variance_population <- var(serie) * (n - 1) / n
# Affichage de la valeur de la variance de la série statistique
cat("La variance de la population est de : ", variance_population, "\n")

```

3. Résultat du code

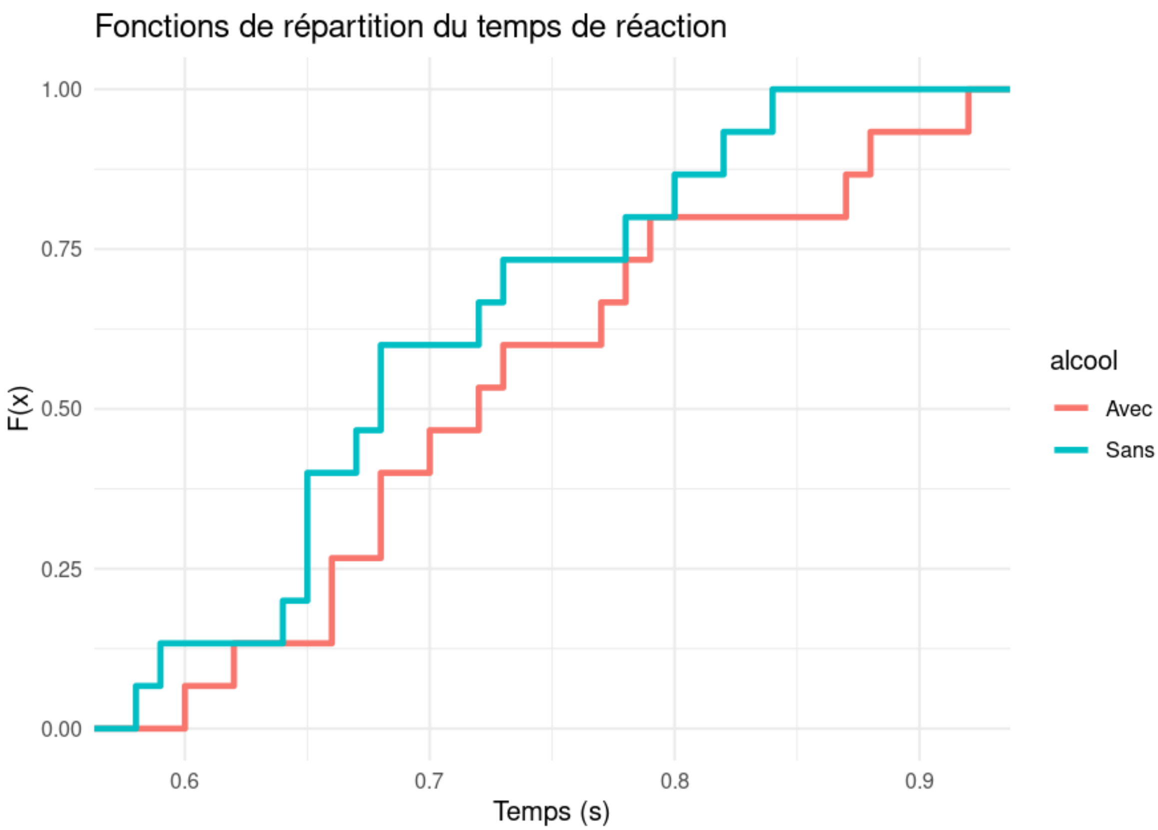
```
> # Calcul de la moyenne
> moyenne <- mean(serie)
> # Affichage du résultat
> cat("La moyenne de la série est :", moyenne, "\n")
La moyenne de la série est : 0.706
> cat("La variance de l'échantillon est de: ", variance_echantillon, "\n")
La variance de l'échantillon est de:  3.2
> # Calcul de la variance de la population (S²)
> variance_population <- var(serie) * (n - 1) / n
> # Affichage de la valeur de la variance de la série statistique
> cat("La variance de la population est de : ", variance_population, "\n")
La variance de la population est de :  0.025744
> 
```

2 *Exercice 2 : Influence de l'alcool sur le temps de réaction*

1. Tracer de la fonction de repartition empirique correspondant aux deux situations (utilisation de simple plot())



1.1 Tracer de la fonction de repartition empirique correspondant aux deux situations (utilisation de simple ggplot2())



2. Test d'hypothèse pour comparer les deux groupes

Il s'agit de montrer l'influence de l'alcool sur le temps de réaction au seuil de risque $\alpha = 5\%$ soit un seuil de confiance de 95%. Pour ce faire on utilisera le test de student car les données sont distribuées dans chaque groupe.

```
sans_alcool <- c(0.68, 0.64, 0.68, 0.82, 0.58, 0.80, 0.72, 0.65, 0.84, 0.73,
                 0.65, 0.59, 0.78, 0.67, 0.65)


# Le vecteur avec_vecteur
avec_alcool <- c(0.73, 0.62, 0.66, 0.92, 0.68, 0.87, 0.77, 0.70, 0.88, 0.79,
                 0.72, 0.60, 0.78, 0.66, 0.68)
```

2.1. Formulation des hypothèses

- Hypothèse nulle (H_0) pour p-value > 0.05 : Les caractères sont normalement distribués (l'alcool n'a pas d'influence sur le temps de réaction.)

- Hypothèse alternative(H_1) pour p-value < 0.05 : Les caractères ne sont pas normalement distribués (l'alcool à une influence sur le temps de réction.)

2.2. Il s'agit de déterminer le risque à prendre pour tirer une conclusion érroné soit $\alpha = 0.05$.

2.3. Vérification des données

```r
# Vérification de la normalité
shapiro_sans_alcool <- shapiro.test(sans_alcool)
shapiro_avec_alcool <- shapiro.test(avec_alcool)

cat("Le taux de normalité pour le groupe sans alcool")
```

```
Le taux de normalité pour le groupe sans alcool
```

```r
shapiro_sans_alcool
```

```

	Shapiro-Wilk normality test

data:  sans_alcool
W = 0.93536, p-value = 0.3275
```

```r
cat("Le taux de normalité pour le groupe avec alcool")
```

```
Le taux de normalité pour le groupe avec alcool
```

```r
shapiro_avec_alcool
```

```

	Shapiro-Wilk normality test

data:  avec_alcool
W = 0.94482, p-value = 0.4468
```

Le test de Shapiro-Wilk montre que les deux groupes *avec alcoool* et *sans alcool* suivent une distribution normale (p $>$ 0.05). Par conséquent l'hypothèse de normalité est respecté pour les deux échantillons.

2.4. Test d'égalité des variances : Déterminer s'il y'a une différence significative entre les variance.

```r
var_test <- var.test(sans_alcool, avec_alcool)
cat("Test dégalité des variance")
```

```
Test dégalité des variance
```

```r
print(var_test)
```

```

	F test to compare two variances

data:  sans_alcool and avec_alcool
F = 0.70037, num df = 14, denom df = 14, p-value = 0.5139
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2351348 2.0861120
sample estimates:
ratio of variances 
         0.7003695 
```

Dans notre cas on p-value ($0.5139 > 0.05$), il indique qu'il n'y a pas de différence significative entre les deux variances. On peut donc supposer l'egalité des variances entre les deux groupes.

2.5. Test de student

```r
# Test de student pour les échantillons indépendants
test_t <- t.test(avec_alcool, sans_alcool, var.equal = TRUE)
cat("Test de student pour les échantillons")
```

```
Test de student pour les échantillons
```

```r
print(test_t)
```

```

	Two Sample t-test

data:  avec_alcool and sans_alcool
t = 1.1923, df = 28, p-value = 0.2432
alternative hypothesis: true difference in means is not equal to 0
```

```
95 percent confidence interval:
 -0.02776448  0.10509781
sample estimates:
mean of x mean of y
0.7373333 0.6986667
```

Résultat du test de student :

- Statistique t : **1.1923**
- Dégré de liberté (n - 2) : **28**
- p-value : **0.2432**
- Intervalle de confiance à 95% : [**-0.02776448 ; 0.10509781**]
- Moyenne du groupe sans alcool : **0.6986667**
- Moyenne du groupe avec alcool : **0.7373333**

Les tests préliminaires de Shapiro–Wilk ont confirmé la normalité des distributions dans les deux groupes ($p > 0.05$), et le test de Fisher a indiqué l'égalité des variances ($p > 0.05$). Le test t de Student pour échantillons indépendants ($t = 1.1923$, $df = 28$, $p = 0.2432$) ne montre aucune différence significative entre les moyennes du groupe avec alcool ($M_{\text{avec alcool}} = 0.7373333$) et du groupe sans alcool ($M_{\text{sans alcool}} = 0.6986667$). Ainsi, la consommation d'alcool ne semble pas avoir d'effet significatif sur la variable mesurée dans cet échantillon.

Par conséquent l'hypothèse H_0 est vérifié. L'alcool n'a pas d'influence significative sur le temps de réaction.

3 *Exercice 3 : Analyse de données pour enfant*

1. Création des vecteur

```r
# Création des vecteurs

#vecteurs individus
Individus = c("Erika", "Célia", "Erik", "Eve", "Paul", "jean", "Adan", "Louis",
              "Jules", "Léo")

#vecteurs Poids
Poids = c(16, 14, 13.5, 15.4, 16.5, 16, 17, 14.8, 17, 16.7)

#vecteurs Taille
Taille = c(100.0, 97.0, 95.5, 101.0, 100.0, 98.5, 103.0, 98.0, 101.5, 100.0)
```

```
#vecteurs sexe
Sexe = c("F", "F", "G","F","G", "G", "G", "G", "G", "G")
```

1.1. Vecteur pour calculer l'âge des individus

```
# Vecteur An
An <- c(3, 3, 3, 4, 3, 4, 3, 3, 4, 3)
# length(An)
# Vecteur Mois

Mois <- c(5, 10, 5, 0, 8, 0, 11, 9, 1, 3)
# length(Mois)
# Calculer l'âge des individus

Age <- round(An + Mois/12, 1)
# Age
```

2. La moyenne des variables (variables quantitatives)

```
#moyenne de la taille
moyenne_taille <- mean(Taille)
moyenne_poids <- mean(Poids)
moyenne_age <- mean(Age)

# Affichage des moyennes
cat("La moyenne des tailles : ", moyenne_taille, "cm\n")
```

```
La moyenne des tailles :  99.45 cm
```

```
cat("La moyenne des poids : ", moyenne_poids, "kg\n")
```

```
La moyenne des poids :  15.69 kg
```

```
cat("La moyenne des âges : ", moyenne_age, "an(s)\n")
```

```
La moyenne des âges :  3.73 an(s)
```

3. Calcul de l'indice de masse corporelle (IMC)

```r
# Taille en mètre
taille_m <- Taille / 100

# Calcul de l'IMC
IMC_echantillon = round((Poids / (taille_m)^2), 2)
# IMC_echantillon

cat("Valeur de l'IMC: \n")
```

```
Valeur de l'IMC:
```

```r
for (i in 1:length(Individus)) {
  cat(Individus[i], " : " ,IMC_echantillon[i], "kg/m²\n")
}
```

```
Erika  :  16 kg/m²
Célia  :  14.88 kg/m²
Erik  :  14.8 kg/m²
Eve  :  15.1 kg/m²
Paul  :  16.5 kg/m²
jean  :  16.49 kg/m²
Adan  :  16.02 kg/m²
Louis  :  15.41 kg/m²
Jules  :  16.5 kg/m²
Léo  :  16.7 kg/m²
```

4. Structure en dataframe

```r
enfant_df <- data.frame(
  Individus = Individus,
  Sexe = Sexe,
  Poids = Poids,
  Taille = Taille,
  Age_complet = Age,
  IMC_echantillon = IMC_echantillon
)

enfant_df
```

```
   Individus Sexe Poids Taille Age_complet IMC_echantillon
```

```
1       Erika    F  16.0  100.0         3.4           16.00
2       Célia    F  14.0   97.0         3.8           14.88
3        Erik    G  13.5   95.5         3.4           14.80
4         Eve    F  15.4  101.0         4.0           15.10
5        Paul    G  16.5  100.0         3.7           16.50
6        jean    G  16.0   98.5         4.0           16.49
7        Adan    G  17.0  103.0         3.9           16.02
8       Louis    G  14.8   98.0         3.8           15.41
9       Jules    G  17.0  101.5         4.1           16.50
10        Léo    G  16.7  100.0         3.2           16.70
```

5. Obtenir les informations sur la fonction plot()

```r
# Informations sur la fonction plot()
?plot()
```

```
démarrage du serveur d'aide httpd ... fini
```

6. Nuage de points du Poids en fonction de la taille

6.1 Installation du package de plotly

```r
# Installation de plotly()
# install.packages("plotly")
# Chargement du package
library(plotly)
install.packages("webshot2")
```

6.2 Calcul du coéfficient de corrélation

```r
# Coéfficient de corrélation entre le Poids et la taille
correlation <- cor(Taille, Poids)
cat("Le coéfficient de corrélation entre la taille et le poids est de : ",
    round(correlation, 3))
```

```
Le coéfficient de corrélation entre la taille et le poids est de :  0.878
```

6.3 Création du graphique

```r
# Ajout de la droite de régression
plot <- plot %>%
  add_trace(
    x = ~Taille,
    y = ~fitted(lm(Poids ~ Taille, data = enfant_df)),
    type = 'scatter',
    mode = 'lines',
    name = "Régression linéaire",
    line = list(color = "red", width = 3, dash = "solid"),
    showlegend = FALSE
  ) %>%
  layout(
    title = list(
      text = '<br><b>Relation Poids-Taille par Sexe</b></br>',
      x = 0.5,
      font = list(size = 18)
    ),
    xaxis = list(
      title = 'Taille (cm)',
      zeroline = FALSE,
      gridcolor = 'lightgrey'
    ),
    yaxis = list(
      title = 'Poids (kg)',
      zeroline = FALSE,
      gridcolor = 'lightgrey'
    ),
    plot_bgcolor = 'white',
    legend = list(x = 0.02, y = 0.98),
    annotations = list(
      list(
        x = 0.02, y = 0.02,
        xref = "paper", yref = "paper",
        text = paste("r =", round(correlation, 3)),
        showarrow = FALSE,
        font = list(size = 14, color = "black"),
        bgcolor = "lightyellow",
        bordercolor = "orange"
      )
    )
  )
```

```
# Affichage du graphique
plot
```

Sortie du code