

Département de Psychologie

## **Guide R**

**(Il y a de l'amour dans R)**

**Daniel de Oliveira Fernandes**

**Christel Borgognon**

**Selina Studer**

**Pascal Wagner-Egger**

## TABLE DES MATIERES

<b>1.</b>	<b>TÉLÉCHARGEMENT ET INSTALLATION DE RSTUDIO.....</b>	<b>5</b>
1.1	INSTALLATION DE R.....	5
1.2	INSTALLATION DE RSTUDIO .....	6
<b>2.</b>	<b>PREMIERS PAS AVEC RSTUDIO.....</b>	<b>7</b>
2.1	VUE D'ENSEMBLE DU PROGRAMME.....	7
2.2	LA SYNTAXE UTILISÉE DANS RSTUDIO.....	8
2.3	TUTORIELS UTILES.....	9
2.4	INSTALLATION DE PACKAGES .....	10
2.5	ACTIVATION DE PACKAGES.....	11
2.6	COMMENT COMMENCER ? – CRÉATION D'UN NOUVEAU PROJET RSTUDIO.....	12
2.7	COMPILEUR UNE COMMANDE .....	13
<b>3.</b>	<b>IMPORTER UNE BASE DE DONNÉES EXISTANTE .....</b>	<b>14</b>
3.1	FICHIER DE DONNÉES SPSS (*.SAV) .....	16
3.2	FICHIER DE DONNÉES EXCEL (*.XLS, *.XLSX) .....	17
3.3	FICHIER DE DONNÉES TEXT (*.CSV) .....	19
3.4	FICHIER DE DONNÉES TEXT (*.TXT) .....	21
<b>4.</b>	<b>LES FONCTIONS .....</b>	<b>23</b>
4.1	FONCTIONS DE BASE.....	23
4.1.1	<i>Demander de l'aide sur une fonction : help {base}.....</i>	23
4.1.2	<i>Création d'un vecteur avec une liste de valeurs : c {base}.....</i>	23
4.1.3	<i>Attacher la base de données importée : attach {base} .....</i>	23
4.2	FONCTIONS DE MANIPULATION DES DONNÉES .....	24
4.2.1	<i>Créer une variable à partir d'une ou plusieurs variable(s) existante(s).....</i>	24
4.2.2	<i>Insérer la nouvelle variable dans la base de données : cbind {base}.....</i>	24
4.2.3	<i>Renommer une variable dans la base de données : names {base} .....</i>	24
4.2.4	<i>Retirer une colonne (ou variable) de la base de données.....</i>	24
4.2.5	<i>Retirer une ligne (ou participant-e) de la base de données.....</i>	24
4.2.6	<i>Supprimer ou Modifier une valeur dans la base de données : edit {base} .....</i>	25
4.2.7	<i>Factoriser une variable nominale : factor {base}.....</i>	25
4.2.8	<i>Nommer les modalités d'une variable nominale : factor {base} .....</i>	25
4.2.9	<i>Recoder des variables .....</i>	26
4.2.10	<i>Scinder la base de données en fonction des modalités d'un facteur .....</i>	27
4.2.11	<i>Observer les valeurs manquantes de la base de données : missmap {Amelia} .....</i>	27
4.2.12	<i>Supprimer des lignes avec valeurs manquantes « NA » : complete.cases {stats} .....</i>	28
4.2.12.1	<i>Supprimer les lignes avec valeurs manquantes de toutes variables confondues.....</i>	28
4.2.12.2	<i>Supprimer les lignes avec valeurs manquantes d'une variable précise .....</i>	28
4.2.13	<i>Remplacer les valeurs manquantes d'une variable par la moyenne .....</i>	28
4.2.14	<i>Standardiser les variables en scores z .....</i>	28
4.2.15	<i>Transformations mathématiques : sqrt {base} ; log10 {base} .....</i>	29
4.3	FONCTIONS DESCRIPTIVES .....	31
4.3.1	<i>Tableau de fréquences : table {base} .....</i>	31
4.3.2	<i>Description d'une variable : summary {base} .....</i>	31
4.3.3	<i>Description générale de la base de données : describe {psych} .....</i>	31
4.3.4	<i>Moyenne : mean {base}.....</i>	32
4.3.5	<i>Écart-type : sd {base}.....</i>	32
4.3.6	<i>Erreur-standard : std.error {plotrix}.....</i>	32
4.3.7	<i>Intervalle de confiance : group.CI {Rmisc} .....</i>	32
4.3.8	<i>Médiane : median {base}.....</i>	33
4.3.9	<i>Intervalle interquartile : IQR {base} .....</i>	33
4.3.10	<i>Statistiques descriptives.....</i>	33

4.3.11	<i>Test de la normalité (Kolmogorov-Smirnov) : lillie.test {nortest}</i>	35
4.3.12	<i>Valeurs extrêmes et aberrantes (ou Outliers)</i>	36
4.3.13	<i>Graphiques</i>	38
4.3.13.1	<i>Modifications des attributs d'un graphique</i>	38
4.3.13.2	<i>Enregistrement d'un graphique en tant qu'image : png {grDevices}</i>	39
4.3.13.3	<i>Distribution de fréquences : barplot {graphics}</i>	39
4.3.13.4	<i>Histogramme : hist {graphics}</i>	40
4.3.13.5	<i>Histogramme avec courbe normale : plotNormalHistogram {rcompanion}</i>	41
4.3.13.6	<i>Boxplot : boxplot {graphics}, ggplot {ggplot2}</i>	41
4.3.13.6.1	<i>Identifier les valeurs extrêmes d'un boxplot</i>	44
4.3.13.7	<i>Diagramme en barres ou Graphique de moyennes</i>	44
4.3.13.8	<i>Graphique d'interaction</i>	47
4.4	<b>FONCTIONS DE STATISTIQUES INFÉRENTIELLES</b>	49
4.4.1	<i>Tableau de décision statistique</i>	49
4.4.2	<i>Khi-carré : chisq.test {stats}</i>	50
4.4.2.1	<i>Tableaux croisés : CrossTable {gmodels}</i>	50
4.4.2.2	<i>Khi-carré d'ajustement : une variable nominale</i>	51
4.4.3	<i>T-test : t.test {stats}</i>	52
4.4.3.1	<i>T-test à échantillon unique (one sample t-test)</i>	53
4.4.3.2	<i>T-test à échantillons (ou groupes) indépendants (independent samples t-test)</i>	53
4.4.3.3	<i>T-test à échantillons appariés ou à mesures répétées (paired samples t-test)</i>	55
4.4.4	<i>ANOVA : aov {stats}</i>	56
4.4.4.1	<i>ANOVA à un facteur intersujet (One-way ANOVA)</i>	56
4.4.4.2	<i>ANOVA à plusieurs facteurs intersujets (Two-way, Three-way, etc. ANOVA)</i>	58
4.4.4.3	<i>Préparation de la base de données pour l'ANOVA à mesures répétées</i>	61
4.4.4.3.1	<i>Préparation pour 1 variable à mesures répétées : gather {tidyverse}</i>	61
4.4.4.3.2	<i>Préparation pour 2 variables (ou plus) à mesures répétées</i>	63
4.4.4.4	<i>ANOVA à mesures répétées (Repeated Measures ANOVA)</i>	65
4.4.4.5	<i>ANOVA mixte (mesures répétées et groupes indépendants)</i>	69
4.4.4.6	<i>MANOVA (Multivariate Analysis of Variance)</i>	75
4.4.4.7	<i>Contrastes orthogonaux (Comparaisons a priori)</i>	78
4.4.5	<i>Tests non paramétriques</i>	80
4.4.5.1	<i>U de Mann-Whitney/W de Wilcoxon : 2 groupes intersujets ou indépendants</i>	80
4.4.5.2	<i>Kruskal-Wallis : n groupes intersujets ou indépendants</i>	81
4.4.5.3	<i>Wilcoxon : 2 facteurs intrasujets ou groupes appariés : wilcox.test {stats}</i>	82
4.4.5.4	<i>ANOVA de Friedman : n facteurs intrasujets ou groupes appariés</i>	83
4.4.6	<i>Corrélations bivariées : cor.test {stats} ; rcorr {Hmisc}</i>	84
4.4.6.1	<i>Corrélation de Bravais-Pearson</i>	85
4.4.6.2	<i>Corrélation de Spearman</i>	87
4.4.6.3	<i>Corrélation partielle</i>	88
4.4.6.4	<i>Corrélation semi-partielle</i>	89
4.4.6.5	<i>Représentation graphique des coefficients de corrélation</i>	91
4.4.6.5.1	<i>Diagramme de dispersion ou Scatterplot</i>	91
4.4.6.5.2	<i>Corrélogrammes</i>	92
4.4.7	<i>Indice de consistance interne – Alpha de Cronbach : alpha {psych}</i>	94
4.4.8	<i>Régression linéaire</i>	96
4.4.8.1	<i>Régression linéaire simple et multiple</i>	96
4.4.8.2	<i>Régression linéaire incluant des variables nominales</i>	98
4.4.8.3	<i>Méthode de sélection Stepwise (Forward, Backward)</i>	100
4.4.8.4	<i>Méthode de sélection hiérarchique</i>	103
4.4.8.5	<i>Analyse des résidus</i>	105
4.4.8.6	<i>Modération</i>	108
4.4.8.6.1	<i>Modération par une variable nominale</i>	108
4.4.8.6.2	<i>Modération par une variable continue</i>	110
4.4.8.7	<i>Médiation</i>	112
4.4.9	<i>Régression logistique</i>	115
4.4.10	<i>Analyse en Composantes Principales (ACP)</i>	119

5.	PRÉSENTATION DES RÉSULTATS SOUS NORMES APA.....	126
5.1	STATISTIQUES DESCRIPTIVES .....	127
5.1.1	<i>Design One-way ANOVA : 1 facteur .....</i>	127
5.1.2	<i>Design Two-way ANOVA : 2 facteurs.....</i>	128
5.2	ANOVA.....	129
5.2.1	<i>ANOVA à un ou plusieurs facteurs indépendants.....</i>	130
5.2.2	<i>ANOVA à mesures répétées et ANOVA mixtes .....</i>	132
5.3	CORRÉLATION .....	135
5.4	RÉGRESSION LINÉAIRE.....	136

# 1. Téléchargement et Installation de RStudio

## 1.1 Installation de R

Pour utiliser RStudio, vous êtes obligé·e d'avoir installé R. Pour cela, allez sur le lien suivant : <https://stat.ethz.ch/CRAN/>

The Comprehensive R Archive Network

**Download and Install R**

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

En fonction de votre système d'exploitation, cliquez sur le lien correspondant.

**Sous Windows :** Lorsque vous cliquez sur « Download R for Windows », vous serez redirigé·e sur une nouvelle page (voir ci-dessous). Cliquez sur « Install R for the first time », puis sur « Download R 3.6.2 for Windows ». Exécutez ensuite la procédure d'installation conventionnelle.

**R for Windows**

Subdirectories:

- [base](#) Binaries for base distribution. This is what you want to [install R for the first time](#).
- [contrib](#) Binaries of contributed CRAN packages (for R >= 2.13.x; managed by Uwe Ligges). There is also information on [third party software](#) available for CRAN Windows services and corresponding environment and make variables.
- [old\\_contrib](#) Binaries of contributed CRAN packages for outdated versions of R (for R < 2.13.x; managed by Uwe Ligges).
- [Rtools](#) Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

**R-3.6.2 for Windows (32/64 bit)**

[Download R 3.6.2 for Windows](#) (83 megabytes, 32/64 bit)

[Installation and other instructions](#)

[New features in this version](#)

**Sous Mac :** Lorsque vous cliquez sur « Download R for (Mac) OS X », vous serez redirigé·e sur une nouvelle page (voir ci-dessous). Cliquez ensuite sur « R-3.6.2.pkg » sous « Latest release ». Exécutez ensuite la procédure d'installation conventionnelle. Veillez à avoir à ce que votre système soit mis à jour (soit OS X 10.11 (El Capitan) ou supérieurs).

**Latest release:**

[R-3.6.2.pkg](#)  
MD5-hash: 837416578abdcfe3efe16b5a95d65ea0  
SHA1-hash: e07a717ab448932fb967472f5f41c28ea9d7506a  
(ca. 77MB)

R 3.6.2 binary for OS X 10.11 (El Capitan) and higher, signed package. Contains R 3.6.2 framework, R.app GUI 1.70 in 64-bit for Intel Macs, Tcl/Tk 8.6.6 X11 libraries and Texinfo 5.2. The latter two components are optional and can be omitted when choosing "custom install", they are only needed if you want to use the `tk` R package or build package documentation from sources.

**Remarque :** Il est possible que les versions de R (ici 3.6.2) soient différentes de celles exemplifiées dans le guide. Cependant, elles ne devraient pas influer sur la marche à suivre pour l'installation.

## 1.2 Installation de RStudio

Pour utiliser RStudio, vous êtes **obligé·e d'avoir préalablement installé R**. Si tel est le cas, allez sur le lien suivant : <https://rstudio.com/products/rstudio/download/>.

	RStudio Desktop Open Source License <b>Free</b>	RStudio Desktop Commercial License <b>\$995 /year</b>	RStudio Server Open Source License <b>Free</b>	RStudio Server Pro Commercial License <b>\$4,975 /year</b> (5 Named Users)
	<a href="#">DOWNLOAD</a> <a href="#">Learn more</a>	<a href="#">BUY</a> <a href="#">Learn more</a>	<a href="#">DOWNLOAD</a> <a href="#">Learn more</a>	<a href="#">BUY</a> <a href="#">Evaluation   Learn more</a>
Integrated Tools for R	✓	✓	✓	✓
Priority Support		✓		✓
Access via Web Browser			✓	✓

Après avoir sélectionné « Download » sous « RStudio Desktop », vous serez redirigé·e sur une nouvelle page. Le site vous recommandera une version de RStudio adaptée à votre système (ci-dessous, un exemple pour Windows 10).

RStudio Desktop 1.2.5033 - [Release Notes](#)

1. Install R. RStudio requires R 3.0.1+.
2. Download RStudio Desktop. Recommended for your system:

 [DOWNLOAD RSTUDIO FOR WINDOWS](#)  
1.2.5033 | 149.83MB

Requires Windows 10/8/7 (64-bit)

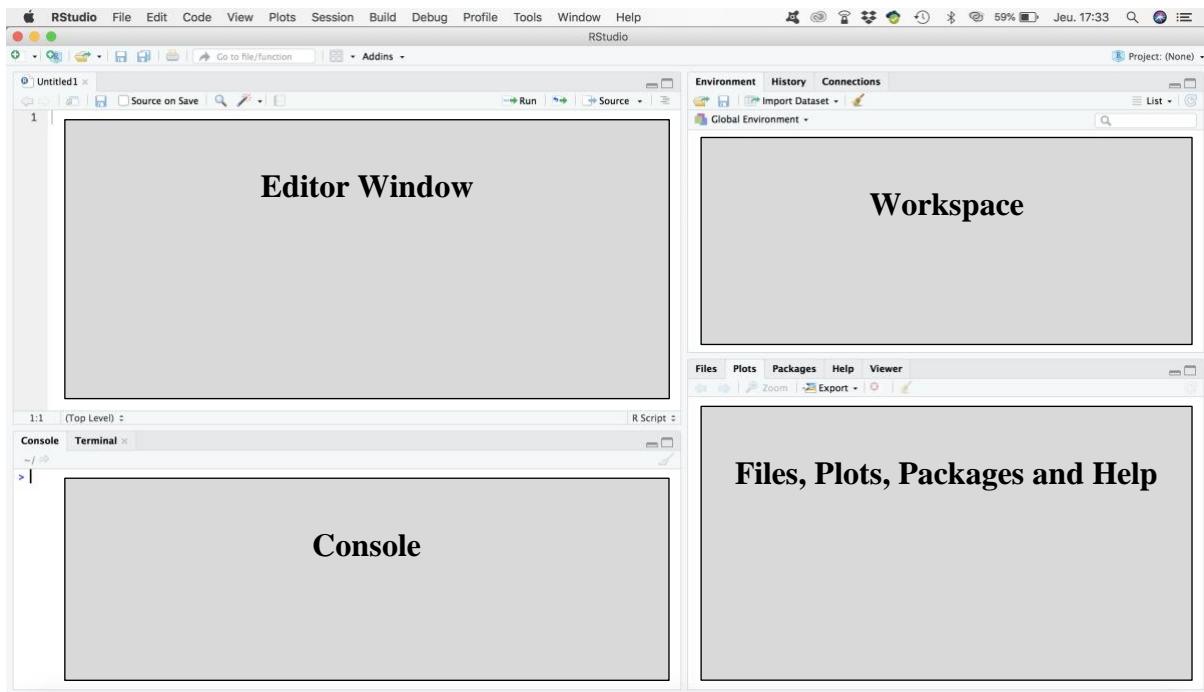
Si la version recommandée par le site ne correspond pas à votre système (ce qui est rare), alors vous pouvez sélectionner manuellement la version correcte.

All Installers			
Linux users may need to <a href="#">import RStudio's public code-signing key</a> prior to installation, depending on the operating system's security policy.			
OS	Download	Size	SHA-256
Windows 10/8/7	 <a href="#">RStudio-1.2.5033.exe</a>	149.83 MB	7fd3bc1b
macOS 10.12+	 <a href="#">RStudio-1.2.5033.dmg</a>	126.89 MB	b67c9875

**En conclusion :** Une fois les deux programmes, **R** et **RStudio**, correctement installés, vous pouvez alors démarrer le programme RStudio en cliquant sur son icône.

## 2. Premiers pas avec RStudio

### 2.1 Vue d'ensemble du programme



- **Editor Window** : Cette fenêtre permet d'afficher vos scripts et l'aperçu de vos bases de données. Il est possible d'afficher plusieurs scripts à la fois tout comme plusieurs aperçus de base de données. Les scripts que vous y aurez ouverts peuvent être enregistrés et réutilisés plus tard. Un script contient tous les commentaires et fonctions utilisées pour transformer, décrire ou analyser vos données.
- **Console** : La console montre le résultat de la compilation d'une fonction. Si vous souhaitez obtenir les statistiques descriptives (ou tout autre chose), vous les obtiendrez dans cette fenêtre. Il est également possible d'y inscrire votre code, mais celui-ci ne sera jamais enregistré pour une utilisation future. Il est alors déconseillé d'écrire vos fonctions dans cette fenêtre (sauf si vous désirez tester rapidement une fonction sans avoir à l'utiliser plus tard). Pour effacer son contenu, appuyez sur Ctrl + L.
- **Workspace** : Le Workspace affiche toutes les variables et les base de données rattachées à votre session de travail (ou projet). Il est possible d'importer des bases de données à partir de la commande « Import Dataset » (voir chapitre concerné pour l'import de données). Vous pouvez ouvrir vos bases de données à partir de cette fenêtre grâce au logo de fenêtre accolées directement à celles-ci. Vous pouvez alterner l'affichage de votre Workspace par le biais d'une liste (sous « List ») ou par le biais d'une grille (sous « Grid »). La grille est préférable car elle permet d'effacer sélectivement une ou plusieurs variable·s ou base·s de données sans avoir à tout effacer. Le balai permet d'effacer l'entièreté de votre Workspace d'un clic.
- **Files, Plots, Packages and Help** : Cette fenêtre affiche de nombreux éléments. Le premier onglet « Files » affiche tous les fichiers présents dans votre projet, ou dossier de travail (directement accessible à l'ouverture en cliquant dessus). Le deuxième onglet « Plots » affiche les graphiques demandés. Le troisième onglet « Packages » affiche tous les packages installés. L'onglet « Help » affiche l'aide pour une fonction spécifiée.

## 2.2 La syntaxe utilisée dans RStudio

La syntaxe diffère d'un programme à l'autre (que cela soit pour R, Matlab ou Python). Il est donc important d'utiliser des syntaxes précises pour une utilité bien spécifique. Dans ce guide, les syntaxes nécessaires seront déjà spécifiées : vous n'aurez donc pas besoin de spécialement les distinguer pour utiliser les fonctions présentes dans ce guide. Cependant, les connaître permet de comprendre l'architecture de certaines commandes plus complexes.

- **Le dièse – # :** Ce caractère permet d'insérer un commentaire dans le script. Le commentaire apparaîtra donc en vert (avec l'affichage par défaut). Ces commentaires sont utiles indiquer ce qui est fait et pourquoi et n'influera jamais sur le code. Il est également possible de créer des sous-chapitres qui peuvent être masqués ou étendus en fonction de l'analyse en cours. En voici deux exemples :

```
## MANOVA sur Beta1/2
MANOVA_Beta12bySegmFCO <- manova(cbind(Beta1, Beta2) ~ GroupBinFCO*VAS_nowSegment,
                                     data=MD_191222_FibroCNPOrtho.inc)
summary(MANOVA_Beta12bySegmFCO, test="Pillai") #Effets globaux
summary.aov(MANOVA_Beta12bySegmFCO)           #Effets individuels
```

Sur l'exemple ci-dessus, nous voyons que les commentaires peuvent à la fois être posés sur une ligne distincte des commandes, mais également à la fin de ces commandes.

```
5 #### - 30.12 - Import et Preparation des donnees #####
6 library(readxl)
7 MD_191222 <- read_excel("MergedData-191222.xlsx")
8 MD_191222_BackUp <- MD_191222
9 View(MD_191222)

5 #### - 30.12 - Import et Preparation des donnees #####
68 #### - 30.12 - Max Clusters (Low/Mid/High Beta) #####
69
70 ## CNP [3]
71 View(MD_191222_CNP.inc.VASdiagnos)
72 View(MD_191222_CNP.inc.VASok)
73 View(MD_191222_CNP.inc)
```

Sur les deux exemples ci-dessus, nous voyons cette fois la création de sous-chapitre au sein même de notre base de données. La première image nous montre le sous-chapitre étendu et la seconde le sous-chapitre réduit. Cela est utile lorsque nous avons de nombreuses analyses à effectuer et de passer de l'une à l'autre de manière efficace. Plusieurs stratégies différentes sont possibles pour créer des sous-chapitres, dont celle proposée dans les exemples.

- **Les parenthèses – ( ) :** En règle générale, les parenthèses comprennent le code nécessaire pour compiler la fonction. Le plus souvent, ce seront les variables, base de données et paramètres qui seront inscrits à l'intérieur de ces parenthèses.
- **Les crochets – [ ] :** Pour sélectionner une liste d'éléments précise, soit par colonne, soit par ligne, les crochets sont utilisés.

```
## Distinction Genre (FEM / HOM)
 DataBaseFEM <- DataBase[Genre=="Femme",]
 DataBaseHOM <- DataBase[Genre=="Homme",]
```

Sur l'exemple ci-dessus, la liste d'éléments sélectionnés sera faite sur la base des *lignes*. La sélection se fait alors *avant* la virgule à l'intérieur des crochets. En l'occurrence, nous souhaitons créer deux bases de données annexes ne prenant pour la première que les lignes dont le genre est égal à *Femme*, et pour la seconde que les lignes dont le genre est égal à *Homme*.

```
## Format reduit de la Base de données
DataBaseMINI <- DataBase[,c("NumID", "Genre", "Condition", "Score")]
DataBaseMINI <- DataBase[,c(1,2,4,10)]
```

Sur l'exemple ci-dessus, la liste d'éléments sélectionnés sera faite sur la base des *colonnes*. La sélection se fait alors *après* la virgule à l'intérieur des crochets. En l'occurrence, nous souhaitons créer une base de données réduite ne prenant en compte qu'un nombre réduit de variables (ou *colonnes*), soit l'ID, le genre, la condition et le score. L'alternative au nom complet est d'indiquer le numéro de la colonne correspondante dans la base de données originale.

- **Le signe dollar - \$ :** Le signe dollar permet de spécifier la base de données à partir de laquelle sera utilisée une variable. D'une certaine manière, le script ne "communique" pas avec la base de données présente dans votre Workspace : les variables que vous spécifiez dans une analyse ne seront pas trouvées si vous ne spécifiez pas la base de données qui la contient ce qui résultera en un message d'erreur.

```
## Moyenne d'age
mean(DataBaseFEM$Age)
```

Sur l'exemple ci-dessus, la base de données à partir de laquelle sera calculée la moyenne d'âge des participant·es est précisée.

```
> ## Moyenne d'age
> mean(Age)
Error in mean(Age) : objet 'Age' introuvable
```

Sur l'exemple ci-dessus, la base de données n'a pas été précisée pour nous permettre de calculer la moyenne, raison pour laquelle la commande s'est soldée par un message d'erreur (« l'objet 'Age' est introuvable »). De plus, la variable « Age » est quasi omniprésente dans toutes bases de données, et il est par ailleurs possible de travailler sur plusieurs bases de données sur un seul projet.

## 2.3 Tutoriels utiles

Ci-dessous, vous sont présentés quelques tutoriels pouvant être utiles pour apprendre à utiliser R et à en comprendre le langage.

- Cyclismo (EN) :
<http://www.cyclismo.org/tutorial/R/types.html>
- R-Tutor (EN) :
<http://www.r-tutor.com/r-introduction>
- ‘R Tutorial: Introduction to R’ (YouTube ; EN) :
<https://www.youtube.com/watch?v=7cGwYMhPDUY>
- Quick-R by DataCamp (EN) :
<https://www.statmethods.net/r-tutorial/index.html>
- Begin'R : Les statistiques avec R (FR) :
<http://beginr.u-bordeaux.fr/index.html#sommaire>

## 2.4 Installation de packages

Pour pouvoir utiliser certaines fonctions sur RStudio, vous aurez besoin d'installer des packages spécifiques (puis de les activer, voir sous-chapitre suivant), soit directement à partir d'une fonction (Solution 1) soit à partir de la commande « Tools » (Solution 2).

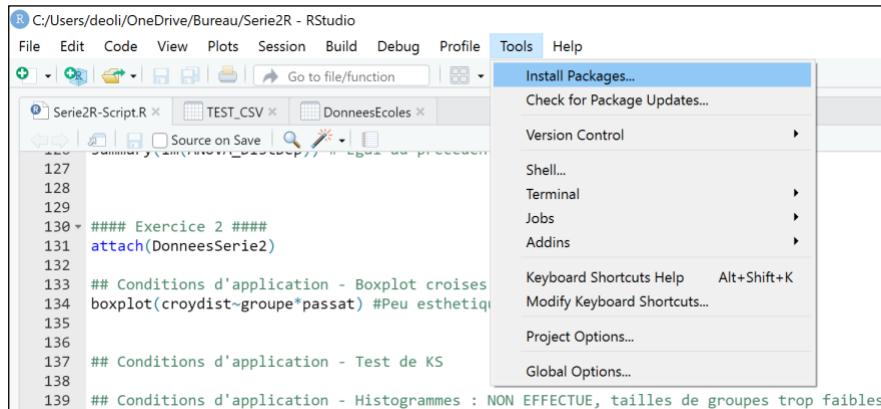
**Remarque :** Les packages nécessaires pour effectuer les fonctions développées tout au long de ce guide sont précisés à l'aide d'accolades (p. ex.: `{nortest}` pour compiler la fonction `lillie.test` et obtenir un test de Kolmogorov-Smirnov avec correction de Lilliefors).

Solution 1 (recommandée) : Nous pouvons directement installer le package au moyen de la fonction `install.package`. Veuillez respecter le code ci-dessous. En **gras**, ce qui est obligatoire et doit rester inchangé quel que soit le package. Ce qui n'est pas en gras est ce que vous pouvez modifier en fonction des packages désirés. Dans le cas présent, nous souhaitons télécharger le package `haven`.

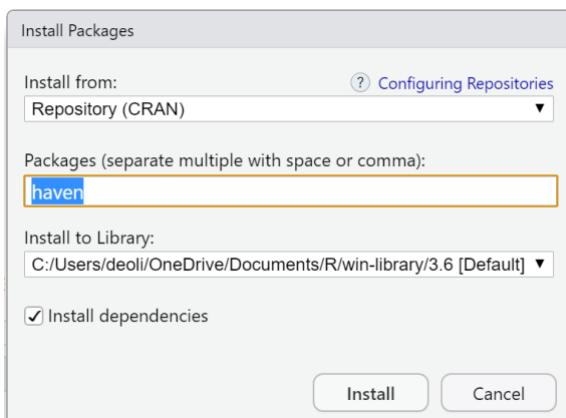
```
install.packages("haven")
```

**Remarque :** Le package est installé, mais n'est pas encore activé pour la session en cours. Pour cela, voir le sous-chapitre suivant.

Solution 2 : Autrement que par le biais de la fonction présentée ci-dessus, vous pouvez suivre le chemin suivant : Tools → Install Packages.



Une fenêtre s'ouvre. Sous « Packages », inscrivez le package que vous souhaitez installer, en l'occurrence « `haven` ». Cliquez ensuite sur « `Install` ».



Dès qu'un package est installé, il n'y a plus nécessité de réitérer le processus : le package est installé pour toujours sur votre logiciel (à moins que vous le désinstalliez). Maintenant que vous avez installé un package, il faut l'activer à chaque session (ou ouverture de RStudio).

## 2.5 Activation de packages

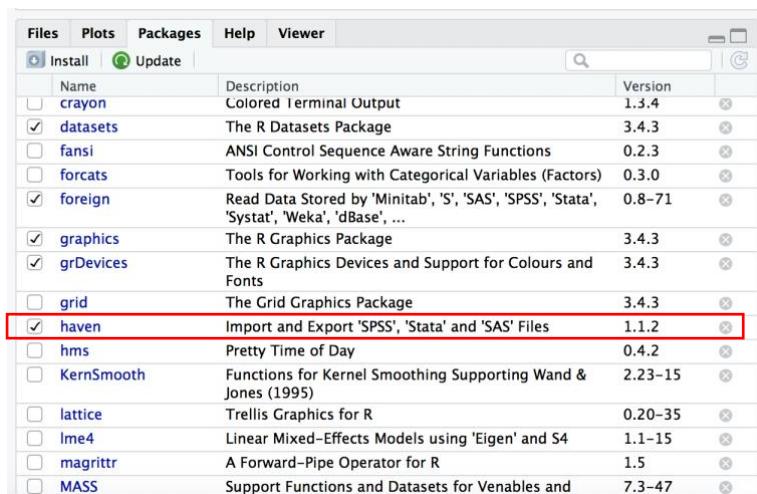
Pour pouvoir utiliser certaines fonctions sur RStudio, vous aurez également besoin d'activer les packages que vous avez précédemment installés, soit directement à partir d'une fonction (Solution 1) soit en cochant le ou les packages désirés depuis la liste des packages disponibles (Solution 2).

**Remarque :** Les packages activés se désactivent à chaque fermeture du programme. Il est alors nécessaire de réactiver ces packages à chaque ouverture pour pouvoir utiliser les fonctions désirées.

Solution 1 (recommandée) : Vous pouvez directement activer les packages au moyen de la fonction `library`. Cette solution est la plus recommandée, car les packages inscrits dans le script restent enregistrés et peuvent être simplement compilés à chaque ouverture de session : il est par exemple possible de créer une section dans votre script avec tous les packages à activer pour vous s'assurer de n'en oublier aucun.

```
library(haven)
```

Solution 2 : Dans la fenêtre située en bas à droite du programme, cliquez sur « Packages ». Dans la liste de packages à votre disposition, cochez ceux qui ont été téléchargés. Cette solution est moins recommandée que la précédente car elle demande de rechercher le package parmi une liste relativement longue de packages présents.



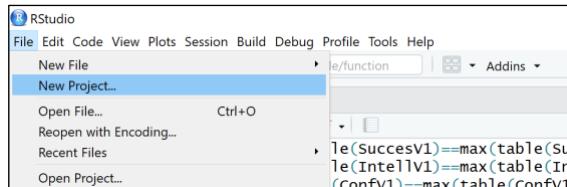
Solution 3 : Il existe également la possibilité de directement accoler le package à la fonction désirée au moyen de `::` (comme exemplifié pour la fonction `describe` se trouvant dans le package `psych`). Grâce à cette méthode, il n'est pas nécessaire de les activer à chaque redémarrage de RStudio puisque vous aurez déjà spécifié de quel package la fonction est tirée.

```
psych::describe(Database)
```

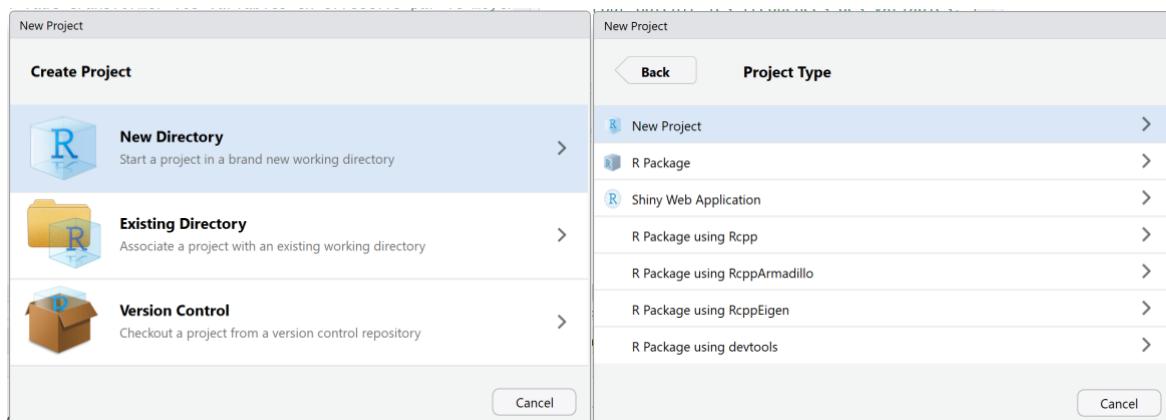
## 2.6 Comment commencer ? – Crédation d'un nouveau projet RStudio

La création d'un projet RStudio vous permet de créer un nouveau dossier comprenant à la fois votre script, vos données et les graphiques. Ainsi, à chaque fois que vous ouvrirez le fichier RStudio depuis ce dossier, le Working Directory (ou *dossier de travail*) sera d'office spécifié.

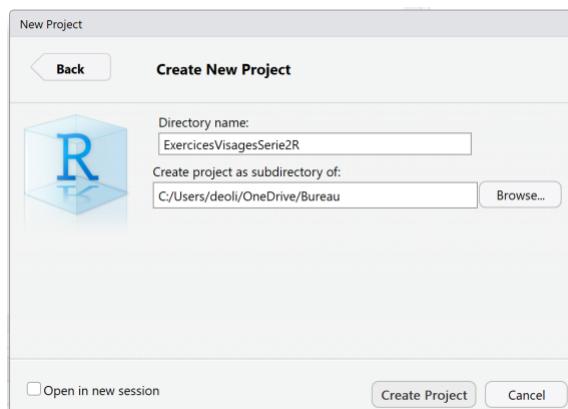
Voici les étapes à suivre : **File → New Project**



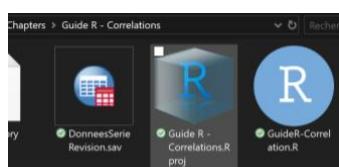
Une fenêtre « New Project » s'ouvre. Cliquez sur **New Directory**, puis sur **New Project**.



Il vous faut spécifier l'emplacement de votre projet RStudio grâce à **Browse** sous « Create project as subdirectory of : ». Sous « **Directory name** », nommez le nouveau dossier qui sera créé à l'emplacement que vous avez spécifié. Pour terminer, appuyez sur **Create Project**.



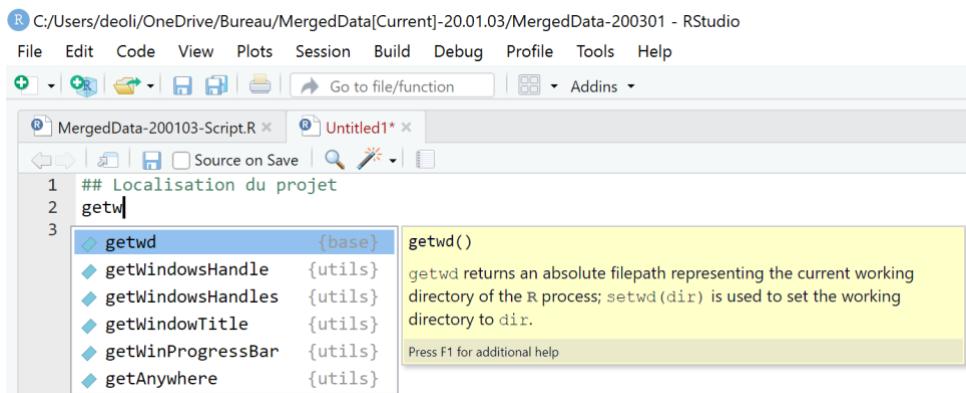
Une nouvelle fenêtre de travail s'ouvre et le Working Directory est alors déjà spécifié. Sous **File**, lancez le nouveau **R Script** et enregistrez-le (celui-ci sera enregistré dans le dossier de votre projet). Si vous souhaitez importer une nouvelle base de données à ce projet, vous pouvez la coller dans ce nouveau dossier. A chaque fois que vous souhaitez travailler sur ce projet, cliquez sur le cube R dans votre dossier de projet.



## 2.7 Compiler une commande

Le programme RStudio est utilisé grâce à des commandes. Une commande peut contenir plusieurs fonctions à la fois (comme lorsque nous souhaitons avoir des statistiques descriptives en demandant à la fois la moyenne et l'écart-type). Pour faire tourner ces commandes, nous devons les *compiler*. Le raccourci au clavier permettant de compiler la commande est soit **Ctrl + Enter** (sous Windows), soit **Cmd + Enter** (sous Mac), à la ligne correspondante (soit seulement **Enter** dans la console, ce qui n'est pas recommandé car ne sera pas sauvegardé).

Exemple : Nous souhaitons utiliser la fonction `getwd`, fonction nous rappelant le dossier dans lequel a été créé notre projet (et où se trouveront nos fichiers, dont le script, si nous les enregistrons).



Vous remarquerez que, lorsque vous écrivez une fonction, un menu déroulant s'affiche et vous propose des fonctions existantes ainsi qu'un bref descriptif. Vous pouvez donc appuyer sur Enter si la fonction que vous voulez utiliser est surlignée en bleu comme dans l'exemple ci-dessus.

A screenshot of the RStudio Console tab. The console output shows:

```
Console Terminal × Jobs ×
C:/Users/deoli/OneDrive/Bureau/MergedData[Current]-20.01.03/MergedData-200301/ ↵
> ## Localisation du projet
> getwd()
[1] "C:/Users/deoli/OneDrive/Bureau/MergedData[Current]-20.01.03/MergedData-200301"
```

The command `getwd()` is highlighted in blue, indicating it has been compiled.

Dans la **Console**, vous obtenez le résultat de votre code après l'avoir compilé. En l'occurrence, le projet se trouve dans le dossier .../*MergedData-200301*, soit le projet que nous avons créé comme indiqué dans le chapitre précédent, lui-même se trouvant dans le dossier .../*MergedData[Current]-20.01.03*, qui lui-même se trouve dans le Bureau.

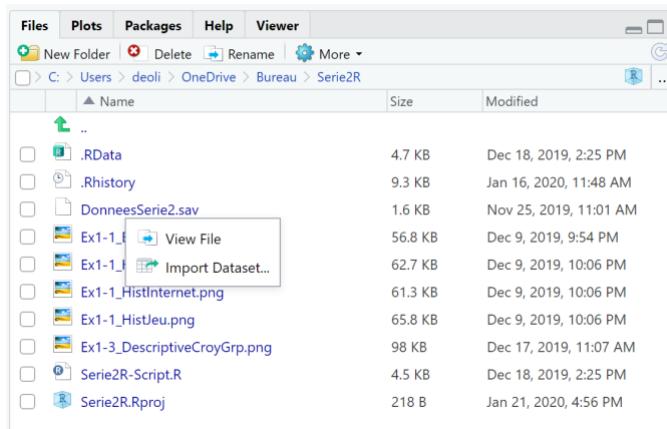
### 3. Importer une base de données existante

Package à activer : {haven} ; {readr}

**Attention :** Le nom des fichiers ne doit contenir ni caractère spéciaux (p. ex.: é) ni espaces.

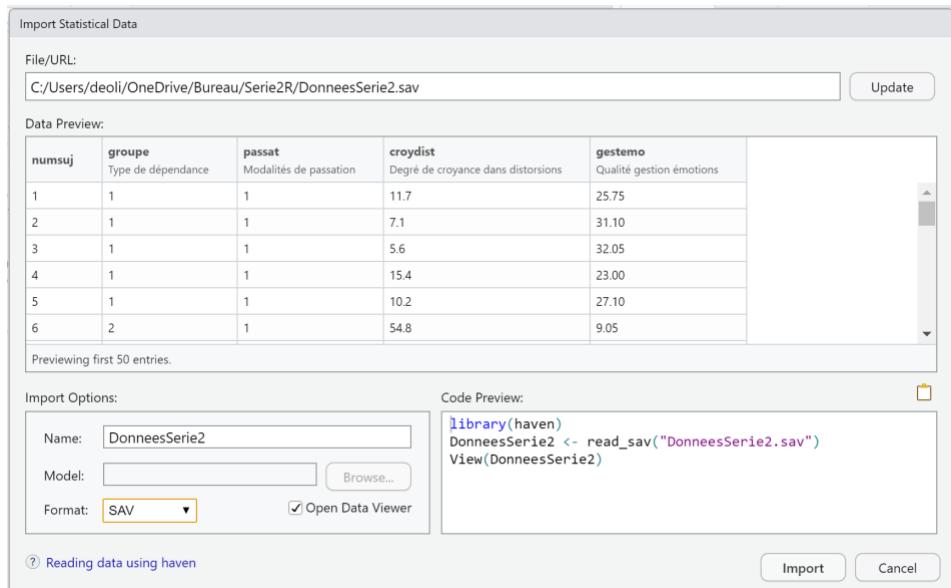
Il existe plusieurs manières d'importer une base de données sur R. L'une d'elle consiste à utiliser la commande appropriée par le biais du script. Une autre consiste à utiliser le menu « Import Dataset » et d'en sélectionner l'extension appropriée. La troisième méthode développée dans ce guide est la plus simple et est recommandée pour gagner du temps sans perdre en efficacité : elle consiste simplement à cliquer sur la base de données présente sous l'onglet « Files ». Quelle que soit la méthode, pour simplifier l'import de données, **il vous sera nécessaire d'insérer la base de données dans le dossier du projet**.

Méthode 1 (recommandée) – Par le biais de l'onglet « Files » : Pour importer une base de données à partir de cet onglet, il faut tout d'abord que votre base de données se trouve dans le projet que vous avez précédemment créé. Pour ce faire, vous pouvez simplement copier/coller votre base de données dans ce dossier (ou directement *Enregistrer sous* et spécifier votre dossier de projet). Cette méthode est possible pour les fichiers aux extensions \*.xls (ou \*.xlsx), \*.sav et \*.csv<sup>1</sup>, entre autres. Elle n'est cependant pas possible pour les fichiers à l'extension \*.txt.



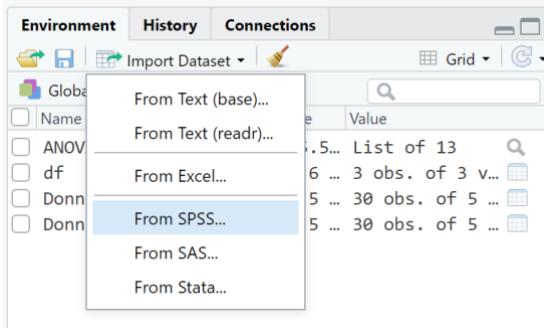
Comme indiqué ci-dessus, en appuyant sur la base de données, un menu apparaît et permet de visualiser le fichier ou de l'importer. En cliquant sur « Import Dataset », vous pourrez directement importer votre base de données sans avoir à spécifier de quelle extension il s'agit (R reconnaîtra de lui-même s'il s'agit d'un fichier Excel ou SPSS).

<sup>1</sup> Une légère subtilité quant à l'import de l'extension .csv est développée au sous-chapitre consacré (*Fichier de données Text (\*.csv)*) (3.3), méthode 2.



Une nouvelle fenêtre « Import Statistical Data » apparaît alors. L'image ci-dessus montre l'exemple avec un fichier de données avec extension \*.sav (SPSS). Les autres extensions montrent des fenêtres quasi identiques. Sous « Code Preview », la commande qu'il aurait été nécessaire d'inscrire sur votre script que R va compiler si vous appuyez sur « Import ».

**Méthode 2 – Par le biais du menu « Import Dataset » :** Cette méthode permet d'inclure les fichiers avec extension \*.txt également. Il vous faudra alors spécifier le programme à partir duquel le fichier a été créé : en l'occurrence Text (pour l'extension \*.txt), Excel (\*.xls, \*.xlsx, \*.csv), SPSS (\*.sav), SAS (\*.sas) ou Stata. Pour connaître la spécificité de chacune des extensions citées précédemment, voir les sous-chapitres suivants.

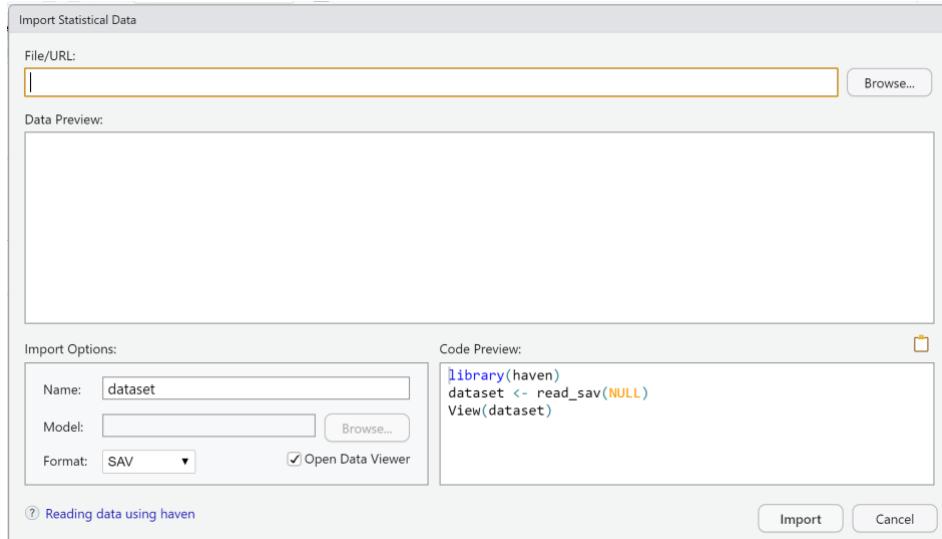


**Méthode 3 – Par le biais d'une commande manuelle :** Cette méthode est celle qu'utilisent "en arrière-plan" les méthodes décrites précédemment. Elle consiste à inscrire manuellement la commande dans le script pour importer votre base de données. Pour connaître les commandes spécifiques pour chacune des extensions citées précédemment, voir les sous-chapitres suivants.

**Remarque :** Pour cette méthode, il est nécessaire d'inscrire les noms exacts (au caractère près).

### 3.1 Fichier de données SPSS (\*.sav)

Méthode 2 : Après avoir sélectionné un import à partir d'une base de données SPSS, la fenêtre ci-dessous apparaît :



En cliquant sur « Browse », vous pouvez parcourir vos dossiers et sélectionner le fichier contenant la base de données SPSS. Si le programme SPSS n'est pas installé sur votre ordinateur, l'icône sera un simple rectangle blanc, mais peut tout de même être sélectionné. Vous aurez alors une prévisualisation de la base de données sur la fenêtre d'import et vous pourrez cliquer sur « Import ».

Méthode 3 : Comme développé précédemment, il est également possible d'inscrire la commande directement dans le script. Dans l'exemple ci-dessous, le fichier de données `DonneesSerie.sav` est présent dans le même dossier que le projet.

```
Database <- read_sav("DonneesSerie.sav")
```

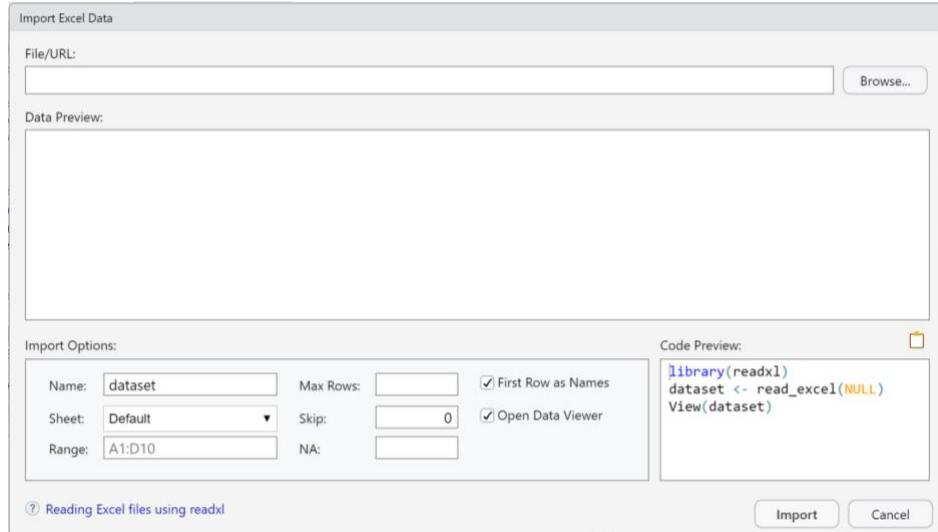
Dans le cas où le fichier de données ne se trouve pas dans le même dossier, il est nécessaire d'inscrire le chemin entier où se trouve ce fichier comme développé dans l'exemple ci-dessous. Nous recommandons de coller votre fichier de données directement *dans* votre dossier (projet).

```
Database <- read_sav("C:/Users/deoli/Bureau/DonneesSerie.sav")
```

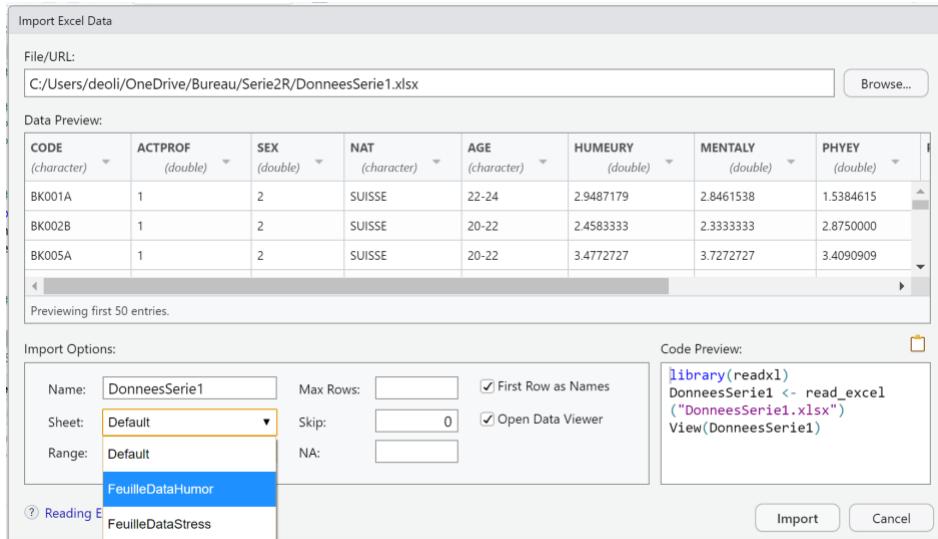
L'élément avant `<-`, soit `Database`, est le nom (sans accents) que vous attribuez à votre base de données importée. Si le fichier de base contient trop de caractères, il est tout à fait possible d'attribuer un nom plus court ou de lui donner un nom plus explicite comme `DonneesVisages`. Ce nom s'affichera alors sur votre Workspace et vous pourrez désormais travailler avec. Si vous omettez cette attribution de nom, la base de données n'apparaîtra alors pas dans votre Workspace (et il sera impossible de travailler sur ces données).

### 3.2 Fichier de données Excel (\*.xls, \*.xlsx)

Méthode 2 : Après avoir sélectionné un import à partir d'une base de données Excel, la fenêtre ci-dessous apparaît :



En cliquant sur « Browse », vous pouvez parcourir vos dossiers et sélectionner le fichier contenant la base de données Excel. Si le programme Excel n'est pas installé sur votre ordinateur, l'icône sera un simple rectangle blanc, mais peut tout de même être sélectionné. Vous aurez alors une prévisualisation de la base de données sur la fenêtre d'import.



Après avoir sélectionné le fichier de données, vous aurez également la possibilité de sélectionner la Feuille (ou *Sheet*) du fichier Excel : si votre fichier ne contient qu'une feuille, il n'y a pas nécessité de sélectionner la feuille car elle sera utilisée par défaut. Si le nom de vos variables est inscrit sur la première ligne de votre fichier de données Excel, maintenez cochée la case « First Row as Names ». Quand tous les paramètres sont correctement sélectionnés, vous pourrez alors cliquer sur « Import ».

**Méthode 3 :** Comme développé précédemment, il est également possible d'inscrire la commande directement dans le script. Dans l'exemple ci-dessous, le **fichier de données** `DonneesSerie.xlsx` est **présent dans le même dossier que le projet**.

```
Database <- read_excel("DonneesSerie.xlsx")
```

Il est également possible qu'un fichier Excel contiennent plusieurs feuilles contenant des données différentes chacune. Dans ce cas, vous pouvez également **sélectionner la feuille de données** sur laquelle vous voulez travailler avec R grâce au paramètre `sheet`.

```
Database <- read_excel("DonneesSerie.xlsx", sheet="FeuilDataHumor")
```

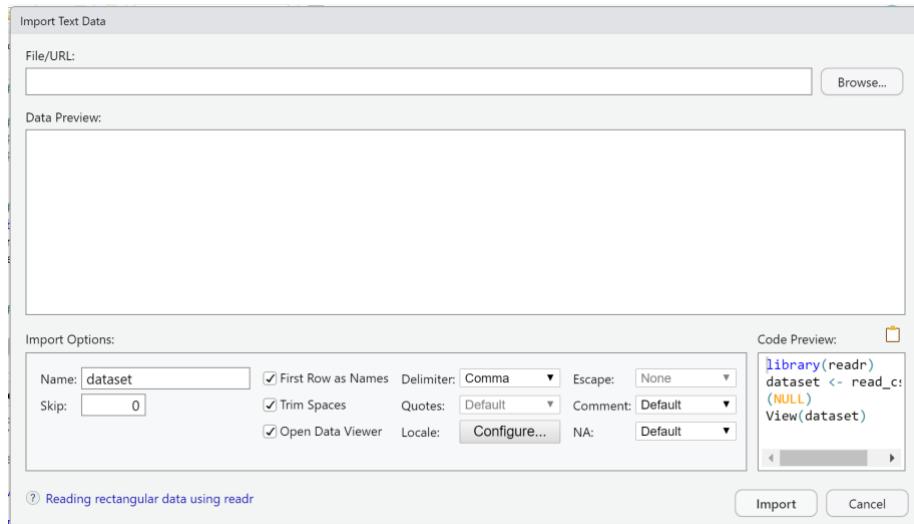
**Dans le cas où le fichier de données ne se trouve pas dans le même dossier**, il est nécessaire d'inscrire le chemin entier où se trouve ce fichier comme développé dans l'exemple ci-dessous. Nous recommandons de coller votre fichier de données directement *dans* votre dossier (projet). Cette commande peut également se combiner avec le paramètre `sheet` présenté précédemment.

```
Database <- read_excel("C:/Users/deoli/Bureau/DonneesSerie.xlsx")
```

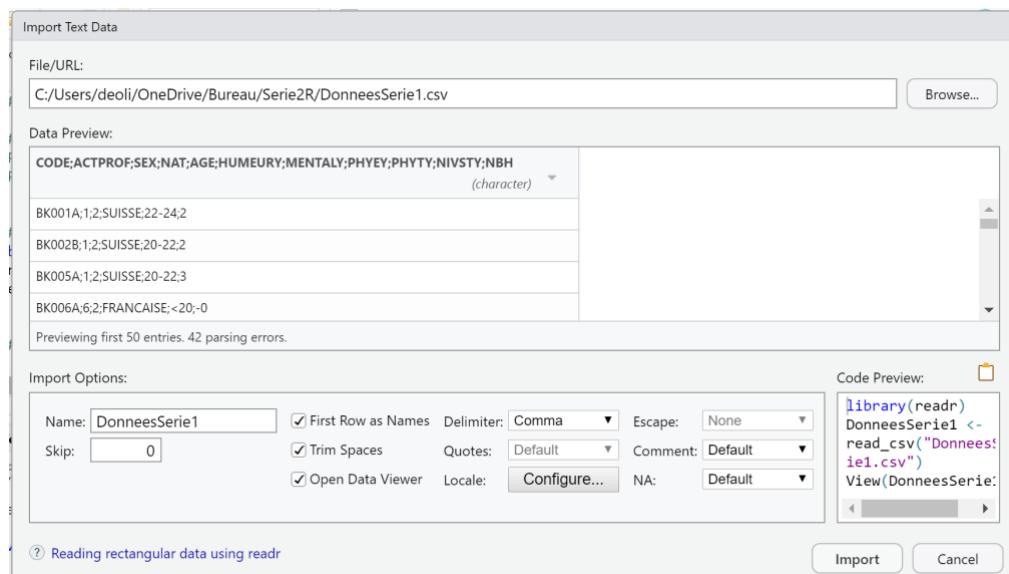
L'élément avant `<-`, soit `Database`, est le nom (sans accents) que vous attribuez à votre base de données importée. Si le fichier de base contient trop de caractères, il est tout à fait possible d'attribuer un nom plus court ou de lui donner un nom plus explicite comme `DonneesVisages`. Ce nom s'affichera alors sur votre Workspace et vous pourrez désormais travailler avec. Si vous omettez cette attribution de nom, la base de données n'apparaîtra alors pas dans votre Workspace (et il sera impossible de travailler sur ces données).

### 3.3 Fichier de données Text (\*.csv)

Méthode 2 : Après avoir sélectionné un import à partir d'une base de données Text (**readr**), la fenêtre ci-dessous apparaît :



En cliquant sur « Browse », vous pouvez parcourir vos dossiers et sélectionner le fichier contenant la base de données \*.csv.



Dans le cas présenté ci-dessus, les données n'ont pas été "comprises" correctement par R. Cela est dû à la délimitation (ou *Delimiter*) incorrecte, en l'occurrence des virgules (ou *Comma*) sélectionnées par défaut par R, alors que nos valeurs sont séparées par des points-virgules.

Data Preview:

CODE (character)	ACTPROF (double)	SEX (double)	NAT (character)	AGE (character)	HUMEURY (character)	MENTALY (character)	PHVEY (character)
BK001A	1	2	SUISSE	22-24	2,95	2,85	1,54
BK002B	1	2	SUISSE	20-22	2,46	2,33	2,88
BK005A	1	2	SUISSE	20-22	3,48	3,73	3,41
RK006A	6	2	FRANCAISE	<20	-0,39	-0,41	-0,89

Previewing first 50 entries.

Import Options:

Name: DonneesSerie1    First Row as Names   Delimiter: Semicolon   Escape: None   Comment: Default   NA: Default

Skip: 0    Trim Spaces   Quotes: Comma   Locale: Semicolon

Open Data Viewer   Tab   Whitespace   Other...

Code Preview:

```
library(readr)
DonneesSerie1 <- read_delim("DonneesSerie1.csv",
",", escape = "auto", na = "NA")
```

Import   Cancel

CSV   DonneesSerie1.csv   2.3 KB

En sélectionnant la délimitation correcte, qui peut différer d'un fichier à l'autre, en l'occurrence par des points-virgules (ou *Semicolon*) dans l'exemple ci-dessus, nous obtenons une prévisualisation correcte de nos données. Les autres options de délimitations proposées sont la tabulation (ou *Tab* ; sur clavier : ↵ ou →) et l'espace simple (ou *Whitespace*).

Si le nom de vos variables est inscrit sur la première ligne de votre fichier de données, maintenez cochée la case « First Row as Names ». Quand tous les paramètres sont correctement sélectionnés, vous pourrez alors cliquer sur « Import ».

**Méthode 3 :** Comme développé précédemment, il est également possible d'inscrire la commande directement dans le script. L'import de données à partir de fichiers \*.csv demande d'inscrire davantage de paramètres dont voici un résumé :

- **Nom des colonnes en première ligne :** Sous `col_names=`, vous pouvez préciser si la première ligne de votre fichier de données correspond au nom de vos colonnes (ou variables). En précisant `col_names=FALSE`, vous indiquerez à R que la première ligne ne correspond pas au nom de vos colonnes (et que par conséquent, la première ligne contient des valeurs). Par défaut, ce paramètre considère que la première ligne est le nom des colonnes (`col_names=TRUE`) : dans un tel cas, vous n'aurez donc pas besoin de le préciser.
- **Délimitation :** La délimitation est à préciser simplement en l'inscrivant entre guillemets (";" pour le point-virgule, " " pour l'espace, "\t" pour la tabulation).

Dans l'exemple ci-dessous, le **fichier de données** `DonneesSerie.csv` est **présent dans le même dossier que le projet**. En l'occurrence, le fichier de données espaces ses valeurs par des tabulations et la première ligne représente le nom des colonnes.

```
Database <- read_delim("DonneesSerie.csv", "\t")
```

Dans ce deuxième exemple, le fichier de données espaces ses valeurs par des points-virgules et la première ligne ne correspond pas au nom des colonnes.

```
Database <- read_delim("DonneesSerie.csv", ";" , col_names = FALSE)
```

**Dans le cas où le fichier de données ne se trouve pas dans le même dossier**, il est nécessaire d'inscrire le chemin entier où se trouve ce fichier comme développé dans l'exemple ci-dessous. Nous recommandons de coller votre fichier de données directement *dans* votre dossier (projet). Cette commande peut également se combiner avec les paramètres présentés précédemment.

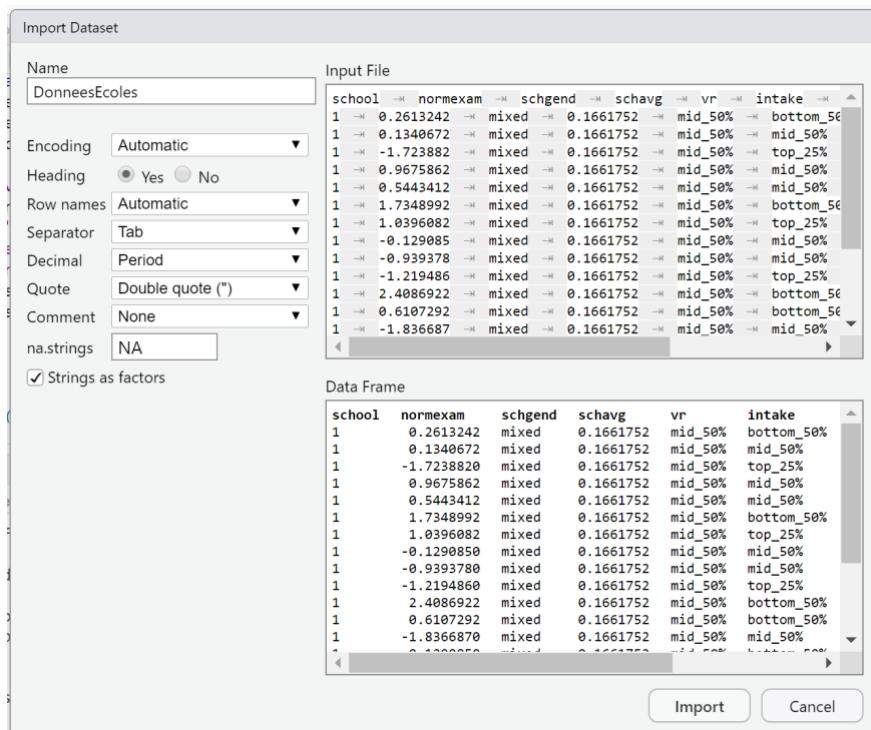
```
Database <- read_delim("C:/Users/deoli/Bureau/DonneesSerie.csv")
```

L'élément avant `<-`, soit `Database`, est le nom (sans accents) que vous attribuez à votre base de données importée. Si le fichier de base contient trop de caractères, il est tout à fait possible d'attribuer un nom plus court ou de lui donner un nom plus explicite comme `DonneesVisages`. Ce nom s'affichera alors sur votre Workspace et vous pourrez désormais travailler avec. Si vous omettez cette attribution de nom, la base de données n'apparaîtra alors pas dans votre Workspace (et il sera impossible de travailler sur ces données).

### 3.4 Fichier de données Text (\*.txt)

**Méthode 2 :** Après avoir sélectionné un import à partir d'une base de données Text (**base**) et choisi le fichier de données \*.txt à importer, la fenêtre ci-dessous apparaît. Elle donne un aperçu du fichier d'origine sous « Input File » en y spécifiant automatiquement la délimitation des valeurs (en l'occurrence, des tabulations sous le signe `→`), et un aperçu de la base de données finale qui sera enregistrée sur votre Workspace.

Sous « Name », il est possible de modifier le nom (sans accent) qui sera attribuée à votre base de données sur R. Le paramètre « Heading » permet de spécifier si la première ligne correspond au nom des colonnes (soit le nom de vos variables). Si le programme ne parvient pas à identifier les bonnes délimitations, alors il est possible de le corriger grâce au paramètre « Separator ».



Méthode 3 : Comme développé précédemment, il est également possible d'inscrire la commande directement dans le script. L'import de données à partir de fichiers \*.txt demande d'inscrire davantage de paramètres dont voici un résumé :

- **Nom des colonnes en première ligne** : Sous `header=`, vous pouvez préciser si la première ligne de votre fichier de données correspond au nom de vos colonnes (ou variables). En précisant `header=FALSE`, vous indiquerez à R que la première ligne ne correspond pas au nom de vos colonnes (et que par conséquent, la première ligne contient des valeurs). En précisant `header=TRUE`, vous indiquerez à R que la première ligne correspond au nom de vos colonnes.
- **Délimitation** : Sous `sep=`, la délimitation est à préciser simplement en l'inscrivant entre guillemets (";" pour le point-virgule, " " pour l'espace, "\t" pour la tabulation).

Dans l'exemple ci-dessous, le **fichier de données** `DonneesSerie.txt` est **présent dans le même dossier que le projet**. En l'occurrence, le fichier de données espace ses valeurs par des tabulations et la première ligne représente le nom des colonnes.

```
Database <- read.table("DonneesSerie.txt", header=TRUE, sep="\t")
```

Dans ce deuxième exemple, le fichier de données espace ses valeurs par des points-virgules et la première ligne ne correspond pas au nom des colonnes.

```
Database <- read.table("DonneesSerie.txt", header=FALSE, sep=";")
```

**Dans le cas où le fichier de données ne se trouve pas dans le même dossier**, il est nécessaire d'inscrire le chemin entier où se trouve ce fichier comme développé dans l'exemple ci-dessous. Nous recommandons de coller votre fichier de données directement *dans* votre dossier (projet). Cette commande peut également se combiner avec les paramètres présentés précédemment.

```
Database <- read.table("C:/Users/deoli/Bureau/DonneesSerie.txt")
```

L'élément avant `<-`, soit `Database`, est le nom (sans accents) que vous attribuez à votre base de données importée. Si le fichier de base contient trop de caractères, il est tout à fait possible d'attribuer un nom plus court ou de lui donner un nom plus explicite comme `DonneesVisages`. Ce nom s'affichera alors sur votre Workspace et vous pourrez désormais travailler avec. Si vous omettez cette attribution de nom, la base de données n'apparaîtra alors pas dans votre Workspace (et il sera impossible de travailler sur ces données).

## 4. Les fonctions

Dans ce chapitre, vous trouverez les fonctions principalement utilisées dans le cours. Chaque fonction est accompagnée d'un petit descriptif. Ce qui se trouve entre les accolades représente le package nécessaire pour pouvoir compiler la fonction. Dans le rectangle contenant la fonction, les parties en **gras** représentent les caractères nécessaires pour lancer la commande (les parties qui ne le sont pas sont celles où sont insérées vos variables ou autres objets et paramètres désirés). En **bleu**, vous sera indiqué l'emplacement où vous insérez vos variables et en **rouge** là où il est nécessaire d'insérer la base de données.

### 4.1 Fonctions de base

#### 4.1.1 Demander de l'aide sur une fonction : `help {base}`

Pour savoir comment s'utilise une fonction dans R vous pouvez utiliser la fonction `help`. Entre parenthèses, écrivez le nom de la fonction sur laquelle vous aimeriez des informations et ensuite compilez-là. Les informations s'affichent dans la fenêtre en bas à droite (onglet « Help »). La fonction est aussi disponible par simple clic dans la fenêtre en bas à droite.

```
help(getwd)
```

#### 4.1.2 Création d'un vecteur avec une liste de valeurs : `c {base}`

Cette fonction combine les valeurs en un vecteur ou une liste. Tous les éléments sont séparés par une virgule.

```
v1 <- c(4, 10, 2)
```

#### 4.1.3 Attacher la base de données importée : `attach {base}`

Comme vu précédemment au chapitre de la syntaxe, le `$` nous permet d'extraire une colonne d'une base de données importée. Si vous attachez la base de données, vous pouvez accéder aux colonnes de celle-ci sans écrire le `$` à chaque fois, mais simplement en la nommant.

```
attach(Database)  
mean(Age) ; sd(Age)
```

Si vous souhaitez travailler sur deux bases de données en même temps sur le même projet, vous devez vous assurer d'avoir attaché *en dernier* la base de données sur laquelle vous travaillez sur le moment.

Situations conflictuelles : Une variable est présente et inscrite de manière identique entre les deux bases de données (p. ex.: `Age` de `DonneesVisages`, `Age` de `DataFootRacisme`). La base de données a été scindée en fonction des modalités d'un facteur faisant que toutes les variables sont présentes à plusieurs reprises.

**Remarque :** Pour les quelques situations conflictuelles citées ci-dessus, la fonction `attach` est peu recommandée et **ne sera pas à utiliser** pour les chapitres suivants.

## 4.2 Fonctions de manipulation des données

### 4.2.1 Crée une variable à partir d'une ou plusieurs variable(s) existante(s)

Il est possible de combiner plusieurs objets (nombres, vecteurs, matrices, base de données) et de les disposer en colonnes. Comme une formule mathématique, une opération peut être effectuée sur vos variables.

**Exemple :** Nous aimerais créer une moyenne globale du jugement de l'intelligence à partir de deux variables déjà existantes : l'intelligence des visages sans lunette (`IntellL`) et l'intelligence des visages avec lunette (`IntellSL`), variables tirées de la base de données `Database`. Nous nommons notre nouvelle variable `IntellGLO`.

```
Database$IntellGLO <- (Database$IntellL + Database$IntellSL)/2
```

### 4.2.2 Insérer la nouvelle variable dans la base de données : `cbind {base}`

Cette fonction permet d'insérer une nouvelle variable dans votre base de données. En réalité, elle lie (ou *bind*) la variable au travers d'une nouvelle colonne (d'où le *cbind*). Dans l'exemple, nous souhaitons insérer l'intelligence globale à notre base de données `Database`.

**Remarque :** La manipulation décrite au chapitre précédent insère déjà votre nouvelle variable `IntellGLO` dans `Database` car le nouveau nom est accolé à votre base de données.

```
Database <- cbind(Database, IntellGLO)
```

### 4.2.3 Renommer une variable dans la base de données : `names {base}`

Cette fonction permet de renommer une variable au sein même de la base de données. Dans l'exemple, nous souhaitons modifier l'ancien nom `SuccesSL` en `Succes_SL`.

```
names(Database)[names(Database) == "SuccesSL"] <- "Succes_SL"
```

### 4.2.4 Retirer une colonne (ou variable) de la base de données

Cette commande permet de retirer une colonne (ou variable) de votre base de données. Dans l'exemple, nous souhaitons retirer la variable `Domain` de notre base de données `Database`.

```
Database$Domain <- NULL #Suppression de la var "Domain"
```

### 4.2.5 Retirer une ligne (ou participant·e) de la base de données

Cette commande est utile si vous souhaitez retirer la ligne d'un·e ou plusieurs participant·es de la base de données (si elle a de nombreuses valeurs extrêmes ou aberrantes par exemple).

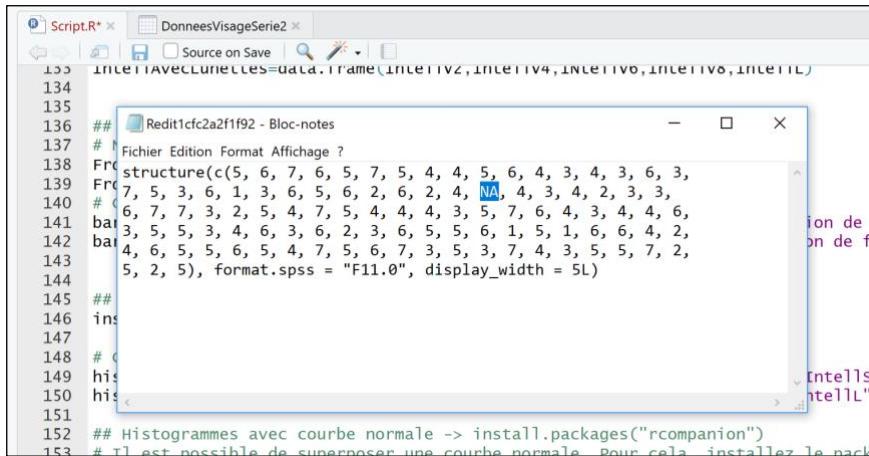
```
Database <- Database[-c(4,5,9),] #Suppression des lignes 4, 5 et 9
```

#### 4.2.6 Supprimer ou Modifier une valeur dans la base de données : edit {base}

Cette fonction est intéressante si vous souhaitez **supprimer** (en inscrivant **NA** à l'emplacement désiré) ou **modifier** une ou plusieurs valeurs d'une variable seule ou même d'une variable tirée d'une base de données. En spécifiant la base de données précédée du dollar et de la variable, cette dernière est modifiée au sein même de la base de données.

```
Database$ConfV1 <- edit(Database$ConfV1)
```

Une fois la commande compilée, une fenêtre Notepad s'ouvre. Vous pouvez alors (1) modifier les valeurs désirées, (2) enregistrer les modifications apportées et (3) fermer la fenêtre.



#### 4.2.7 Factoriser une variable nominale : factor {base}

Il est possible qu'une variable nominale ne soit pas correctement considérée comme un facteur. Cela est notamment le cas lorsque des données sont importées. R considère alors cette variable nominale comme une variable quantitative ou ordinaire : comme si la variable *Groupe* était tirée d'une échelle où les réponses des participant·es sont 1, 2 ou 3.

```
Database$Groupe <- factor(Database$Groupe)
```

Cette fonction permet une factorisation rapide. Cependant, aucun nom n'est attribué à chacune des modalités (hormis des [1], [2], etc.).

#### 4.2.8 Nommer les modalités d'une variable nominale : factor {base}

La fonction décrite au point précédent permet une factorisation rapide des variables nominales non considérées comme facteurs par R. Néanmoins, aucun nom est attribué à chacune modalité de notre nouveau facteur. Nommons cette fois les modalités de la variable *Groupe* par [1] = Footballeur, [2] = Arbitre et [3] = Fan.

**Attention :** Veillez à inscrire dans `levels=c(...)` les niveaux correspondant aux noms respectifs des modalités dans `labels=c(...)`. N'exécutez pas cette commande à deux reprises, car les modalités disparaîtront.

```
Database$Groupe <- factor(Database$Groupe,  
                           levels=c(1,2,3),  
                           labels=c("Footballeur","Arbitre","Fan"))
```

#### 4.2.9 Recoder des variables

Fonctions et packages nécessaires : `recode {car}`, `{data.table}` ; `recode_factor {tidyverse}`

Ces fonctions sont intéressantes pour recoder différemment des valeurs attribuées à des catégories de réponse (au lieu de les modifier une par une). Elles permettent de manipuler des étiquettes déterminées à vos données, mais également de créer des groupes en fonction de valeurs déjà présentes dans votre base de données. Ces recodages sont généralement effectués sur des variables nominales, mais peuvent tout à fait être réalisés sur des variables ordinaires et quantitatives.

Exemples : Nous souhaitons former des classes d'âge (variable ordinaire) à partir de l'âge des participant·es, créer des modalités d'une variable nominale (participant·es jeunes et participant·es âgé·es), recoder un ou plusieurs items dans le sens inverse pour créer une mesure globale à partir de plusieurs items d'une échelle (p. ex.: degré d'émotions positives mesurées grâce à 10 sous-échelles ; voir chapitre *Indice de consistance interne* (4.4.7)).

Situation 1 : Nous souhaitons **recoder des items** qui mesurent des émotions négatives, pour faire un score moyen d'émotion positives. En l'occurrence, nos 4 items d'émotions négatives sont mesurés de 0 (niveau bas d'émotions négatives) à 5 (niveau élevé d'émotions négatives) : nous souhaitons que ce soit l'inverse (0 sera transformé en un haut niveau d'émotions négatives, donc un score bas d'émotions positives ; 5 sera transformé en un bas niveau d'émotions négatives, donc un score bas d'émotions positives ; nous allons donc transformer 0 en 5, 5 en 0, 1 en 4, 4 en 1, 2 en 3 et 3 en 2 (si votre variable compte . Voici l'exemple sur la variable EN03. Créez une nouvelle variable, ici nommée EN03inv, et qui sera automatiquement insérée dans votre base de données Database, afin de garder la variable originale.

```
Database$EN03inv <- car::recode(Database$EN03,  
                                "0=5;1=4;2=3;3=2;4=1;5=0")
```

```
> EN03 #Item original (ordre inverse)  
[1] 1 2 1 0 3 1 2 4 1 3 0 1 1 1 0 1 1 3 3 1 3 5 0 2 5 1 1 2 3 4 1 3 3 1 2 0 0 2 5 3 2 2 2 3  
[45] 1 0 1 3 2 3 2 0 3 0 3 2 0 3 2 1 0 1 1 1 2 3 1 1 1 0 4 1 1 3 1 2 0 3 1 3 5 1 1 1 1 1 1 1  
> EN03inv <- recode(EN03, "0=5;1=4;2=3;3=2;4=1;5=0")  
> EN03inv #Item recode (ordre corrigé)  
[1] 4 3 4 5 2 4 3 1 4 2 5 4 4 4 5 4 4 2 2 4 2 0 5 3 0 4 4 3 2 1 4 2 2 4 3 5 5 3 0 2 3 3 3 2  
[45] 4 5 4 2 3 2 3 5 2 5 2 3 5 2 3 4 5 4 4 4 4 3 2 4 4 4 5 1 4 4 2 4 3 5 2 4 2 0 4 4 4 4 4 4
```

Situation 2 : Nous souhaitons **séparer un échantillon selon leur âge en trois groupes d'âge distincts**, le 1<sup>er</sup> groupe (nommé « Jeune ») allant de 0 à 17 ans, le 2<sup>ème</sup> groupe (nommé « Adulte ») de 18 à 59 ans et le 3<sup>ème</sup> groupe (nommé « Agé ») de 60 à 80 ans.

```
Database$Gr_age <- recode(Database$age, "0:17='Jeune';  
18:59='Adulte'; 60:80='Age'")
```

Vous pouvez vous assurer que vos groupes ont correctement été formés en demandant un tableau de fréquences des modalités de cette nouvelle variable en compilant la commande `table(Database$Gr_age)`.

Situation 3 : Nous souhaitons **regrouper plusieurs modalités d'un facteur en une seule et unique modalité**, tout en maintenant d'autres modalités comme elles l'étaient initialement. Cela peut être le cas lorsque vos données contiennent des catégories ne comportant que très peu d'individus contrairement à d'autres catégories qui en comporte plus. Cette fonction maintient

les valeurs manquantes de la variable originale. Dans cet exemple, nous passons de dix catégories déséquilibrées en nombre d'individus à seulement trois catégories de taille égale.

```
Database$EtudeDomGRP <- recode_factor(Database$EtudeDom,
  'Psychologie'='Lettres', 'Philosophie'='Lettres',
  'Medecine'='Sciences', 'Chimie'='Sciences', .default='Autres')
```

#### 4.2.10 Scinder la base de données en fonction des modalités d'un facteur

Cette commande permet de séparer une importante base de données en fonction des modalités d'un facteur. Cela permet d'avoir des informations distinctes sur divers groupes en travaillant indépendamment des autres modalités de la variable.

Exemple : Nous souhaitons avec une première base de données avec *uniquement* les participantes et une seconde base de données avec *uniquement* les participants.

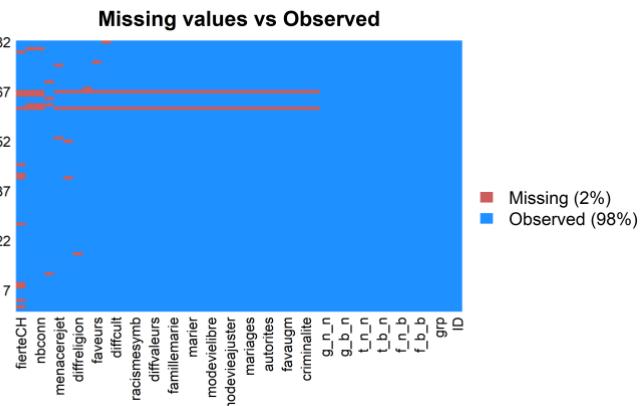
```
attach(Database)
Database_FEM <- Database[Sexe=="Femme", ]
Database_HOM <- Database[Sexe=="Homme", ]
detach(Database)
```

La commande ci-dessus crée deux bases de données distinctes (l'une pour les femmes, l'autre pour les hommes). Dans les crochets, l'élément avant la virgule se rapporte aux colonnes de notre base de données, ici la variable `Sexe`, et l'élément après la virgule aux lignes (comme il n'y a rien à spécifier, nous pouvons laisser un espace vide). Si vous avez des **valeurs manquantes** dans votre facteur de séparation (ici `Sexe`), référez-vous au point 4.2.12.

#### 4.2.11 Observer les valeurs manquantes de la base de données : missmap {Amelia}

La commande suivante est intéressante pour observer, de manière graphique, l'état global de notre base de données au niveau des variables manquantes. Elle permet également de connaître le pourcentage de valeurs manquantes de la base de données. Dans l'exemple ci-dessous, nous constatons que les valeurs manquantes représentent 2% de toute la base de données `Database`.

```
missmap(Database, main = "Missing values vs Observed")
```



## 4.2.12 Supprimer des lignes avec valeurs manquantes « NA » : `complete.cases` {stats}

Si vous avez des valeurs manquantes (inscrites NA = Not Available dans la base de données) dans votre facteur de séparation (4.2.10), la nouvelle base de données sera partiellement corrompue. Pour pallier ce problème, une commande supplémentaire est nécessaire. Vous avez deux choix :

### 4.2.12.1 Supprimer les lignes avec valeurs manquantes de toutes variables confondues

Dans le cas où vous souhaitez supprimer toutes les lignes comprenant au moins une valeur manquante, quelle que soit la variable en question. Nous souhaitons, pour cela, créer une nouvelle base de données sans valeurs manquantes appelée `Database.noNA`.

```
Database.noNA <- Database[complete.cases(Database), ]
```

### 4.2.12.2 Supprimer les lignes avec valeurs manquantes d'une variable précise

Dans le cas où vous souhaitez supprimer toutes les lignes comprenant une valeur manquante, mais uniquement pour une variable précise. Dans cet exemple, uniquement les lignes avec valeurs manquantes de la variable `Age`.

```
Database.AgenoNA <- Database[complete.cases(Database[, c("Age")]), ]
```

## 4.2.13 Remplacer les valeurs manquantes d'une variable par la moyenne

Dans le cas où de trop nombreuses valeurs sont manquantes et afin d'éviter de perdre trop de données, il est possible de remplacer toutes valeurs manquantes d'une variable par la moyenne.

```
Database$Age[is.na(Database$Age)] <- mean(Database$Age, na.rm=TRUE)
```

## 4.2.14 Standardiser les variables en scores z

La commande telle qu'elle est présentée ci-dessous permet de standardiser vos variables en transformant leurs valeurs par des scores z, tout en les insérant directement dans votre base de données. Dans cet exemple, nous souhaitons standardiser la variable `pb_soc`.

```
Database$pb_soc.z <- scale(Database$pb_soc)
```

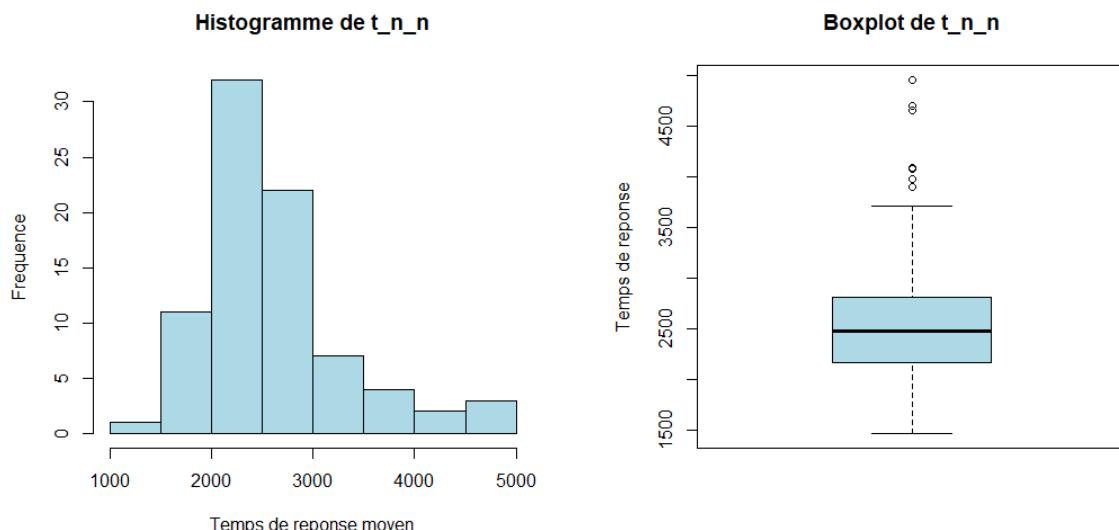
caux	pb_sco Pbs scolaires	pb_soc Pbs sociaux	pb_fam Pbs familiaux	pb_leg Pbs légaux	Conso_Dro Gravité conso. drogue	Conso_OH Gravité conso. alcool	scoreZ_pb_soc
7	8	5	5	9	6	0	0.07868739
0	3	1	1	1	2	1	-1.86882560
2	6	6	6	3	4	0	0.56556564
1	5	3	8	1	7	3	-0.89506911
5	7	7	7	9	9	8	1.05244389
8	6	6	6	8	9	0	0.56556564

#### 4.2.15 Transformations mathématiques : `sqrt {base}` ; `log10 {base}`

Si vous ne pouvez enlever les valeurs extrêmes ("outliers") parce qu'elles sont trop nombreuses (vous perdriez ainsi un trop grand nombre d'observations) ou que la distribution est fortement asymétrique, vous pouvez essayer des transformations mathématiques de tous les scores de votre variable, comme la racine de x, l'inverse de x ( $1/x$ ) ou encore le logarithme (p. ex.: en base 10) de x, afin de voir si la nouvelle distribution obtenue est plus proche de la normale. Pour ce faire, il suffit simplement d'effectuer l'opération désirée sur toutes les valeurs de la variable :

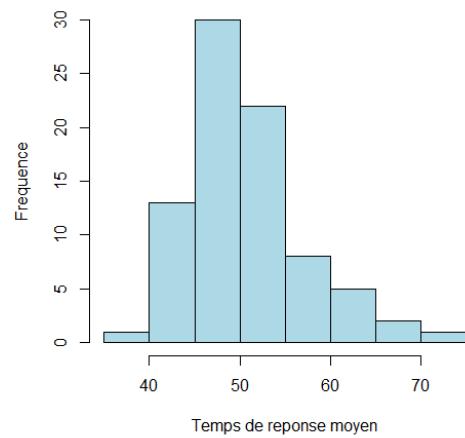
```
Database$tnn_RC <- sqrt(Database$t_n_n) #SQRT = Square Root  
Database$tnn_LOG <- log10(Database$t_n_n)  
Database$tnn_INV <- 1/(Database$t_n_n)
```

Prenons l'exemple de la variable `t_n_n`, tirée de la base de données `Database`, représentant le temps de réaction dans une tâche informatisée. Sa distribution est fortement asymétrique.

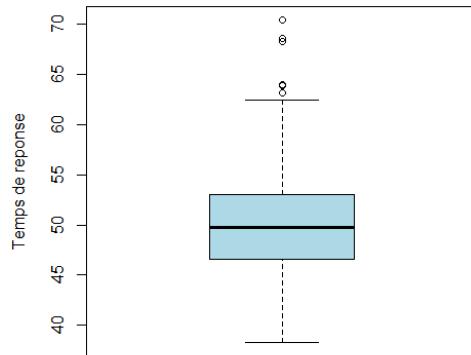


La meilleure des trois transformations est l'inverse, la distribution devient plus normale que l'originale :

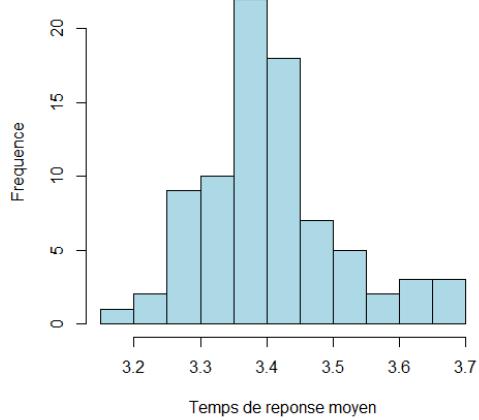
**Histogramme de Racine de t\_n\_n**



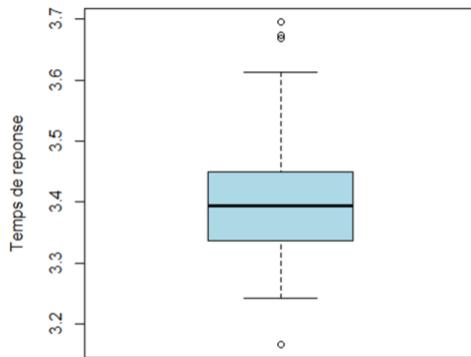
**Boxplot de Racine de t\_n\_n**



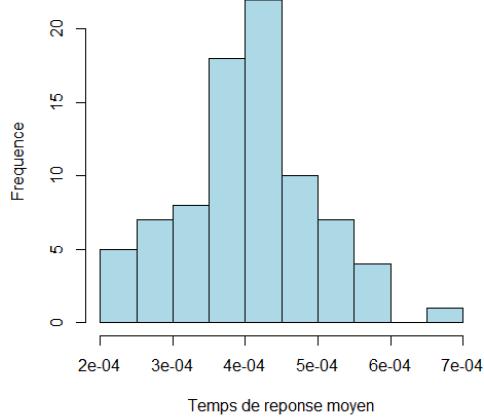
**Histogramme de Logarithme de t\_n\_n**



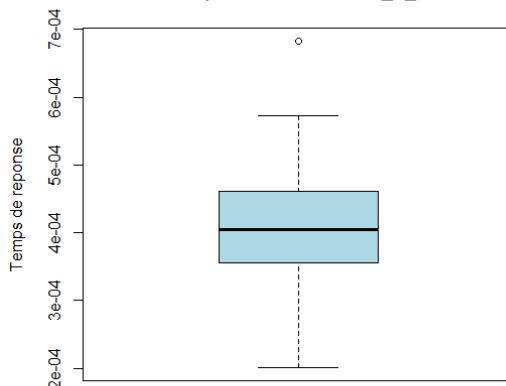
**Boxplot de Logarithme de t\_n\_n**



**Histogramme de Inverse de t\_n\_n**



**Boxplot de Inverse de t\_n\_n**



## 4.3 Fonctions descriptives

### 4.3.1 Tableau de fréquences : table {base}

Cette commande permet d'obtenir un tableau de fréquences absolues (nombres) d'un facteur à plusieurs modalités ou de toute autre variable (ordinale et quantitative). Le paramètre `useNA="ifany"` permet de lister également toutes les valeurs manquantes présentes dans la variable. Ce paramètre peut être retiré si vous ne désirez pas lister les valeurs manquantes.

```
table(Database$Groupe, useNA="ifany")
```

Footballeurs	Arbitres	Fans
43	17	22

Il est également possible d'obtenir un tableau de fréquences relatives (pourcentages) en effectuant une opération sur la table de fréquences.

```
table(Database$Groupe, useNA="ifany") * 100/length(Database$Groupe)
```

Footballeurs	Arbitres	Fans
52.43902	20.73171	26.82927

Si vous souhaitez combiner fréquences absolues et fréquences relatives dans un seul tableau :

```
cbind(table(Database$Groupe, useNA="ifany"),
      table(Database$Groupe, useNA="ifany")
      * 100/length(Database$Groupe))
```

	[,1]	[,2]
Footballeurs	43	52.43902
Arbitres	17	20.73171
Fans	22	26.82927

### 4.3.2 Description d'une variable : summary {base}

Cette fonction produit un **résumé succinct** d'une variable. Une fonction avec davantage d'informations est développée au prochain paragraphe. Dans l'exemple, nous souhaitons un résumé succinct de la variable `Intell1SL` tirée de la base de données `Database`.

```
summary(Database$Intell1SL)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.000	3.750	4.250	4.252	4.750	6.000

### 4.3.3 Description générale de la base de données : describe {psych}

Cette fonction permet d'obtenir les statistiques descriptives les plus utiles de toute la base de données. Dans l'exemple, nous souhaitons un résumé de la base de données `Database`.

```
describe(Database)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
Numparticipant	1	103	52.00	29.88	52	52.00	38.55	1	103	102	0.00	-1.24	2.94
Série	2	103	1.50	0.50	1	1.49	0.00	1	2	1	0.02	-2.02	0.05
EthnieVisages	3	103	1.52	0.50	2	1.53	0.00	1	2	1	-0.10	-2.01	0.05
SuccesV1	4	103	4.28	1.15	4	4.28	1.48	1	7	6	-0.02	-0.15	0.11
Intell1V1	5	103	4.36	1.09	4	4.39	1.48	1	6	5	-0.33	-0.25	0.11
ConfV1	6	102	4.52	1.58	5	4.56	1.48	1	7	6	-0.22	-0.80	0.16

#### 4.3.4 Moyenne : mean {base}

Cette fonction permet de connaître la **moyenne** d'une variable. L'argument `na.rm` doit être `TRUE` si la base de données inclut (ou non) des valeurs manquantes (`NA = Not Available`). Dans l'exemple, nous souhaitons la moyenne de `IntellSL` tirée de la base de données `Database`.

```
mean(Database$IntellSL, na.rm=TRUE)
```

#### 4.3.5 Écart-type : sd {base}

Cette fonction permet de connaître l'**écart-type** (ou *standard deviation, SD*). L'argument `na.rm` doit être `TRUE` si la base de données inclut (ou non) des valeurs manquantes. Dans l'exemple, nous souhaitons l'écart-type de `IntellSL` tirée de la base de données `Database`.

```
sd(Database$IntellSL, na.rm=TRUE)
```

#### 4.3.6 Erreur-standard : std.error {plotrix}

Cette fonction permet de connaître l'**erreur standard** (ou *standard error, SE*). L'argument `na.rm` doit être `TRUE` si la base de données inclut (ou non) des valeurs manquantes. Dans l'exemple, nous souhaitons l'erreur standard de `IntellSL` tirée de la base de données `Database`.

```
std.error(Database$IntellSL, na.rm=TRUE)
```

#### 4.3.7 Intervalle de confiance : group.CI {Rmisc}

Cette fonction permet de connaître l'**intervalle de confiance** désiré (sous `ci=`, l'intervalle de confiance à 95% est généralement utilisée, mais peut également être de 90% ou 99%).

Situation 1 : Nous souhaitons un intervalle de confiance donné d'**une seule variable quantitative** indépendamment d'autres groupes (d'où `~1`). Dans l'exemple, nous souhaitons l'intervalle de confiance à 95% de `IntellSL` tirée de la base de données `Database`.

```
group.CI(IntellSL~1, data=Database, ci=0.95)
```

	IntellSL.upper	IntellSL.mean	IntellSL.lower
1	4.406859	4.252427	4.097995

Situation 2 : Nous souhaitons un intervalle de confiance donné d'**une variable quantitative** en fonction des modalités d'**un facteur**. Dans l'exemple, nous souhaitons l'intervalle de confiance à 99% de `IntellSL` en fonction des modalités du facteur `Genre`, toutes deux tirées de la base de données `Database`.

```
group.CI(IntellSL ~ Genre, data=Database, ci=0.99)
```

	Sexe	IntellSL.upper	IntellSL.mean	IntellSL.lower
1	Femmes	4.482205	4.233108	3.984011
2	Hommes	4.682140	4.259615	3.837091

Le tableau ci-dessus permet d'avoir les informations suivantes : la valeur supérieure de l'intervalle sous « `.upper` », la moyenne sous « `.mean` », la valeur inférieure de l'intervalle sous « `.lower` ». Ces informations peuvent être nécessaires afin de créer des graphiques dont les barres d'erreur sont les intervalles de confiance.

#### 4.3.8 Médiane : median {base}

Cette fonction permet de connaître la **médiane**. L'argument `na.rm` doit être `TRUE` si la base de données inclut (ou non) des valeurs manquantes. Dans l'exemple, nous souhaitons la médiane de `IntellSL` tirée de la base de données `Database`.

```
median(Database$IntellSL, na.rm=TRUE)
```

```
[1] 4.25
```

#### 4.3.9 Intervalle interquartile : IQR {base}

Cette fonction permet de connaître l'intervalle interquartile. L'argument `na.rm` doit être `TRUE` si la base de données inclut (ou non) des valeurs manquantes. Dans l'exemple, nous souhaitons l'intervalle interquartile de `IntellSL` tirée de la base de données `Database`.

```
IQR(Database$IntellSL, na.rm=TRUE)
```

```
[1] 1
```

#### 4.3.10 Statistiques descriptives

Fonctions et package nécessaires : `group_by`, `summarise`, `%>% {plyr} puis {dplyr}`

**Important :** Activez en premier le package `plyr`, puis le package `dplyr` afin de ne rencontrer aucun problème dans les commandes. Dans le cas contraire, vous obtiendrez un message d'erreur qui ne peut être corrigé qu'en redémarrant le logiciel.

La commande suivante permet d'obtenir un tableau avec les statistiques descriptives désirées en fonction d'un ou plusieurs facteurs. Cette commande est plus simple et lisible que les fonctions `by` et `aggregate`. La commande comprend plusieurs éléments dont voici l'ordre :

- **Précision des facteurs** : Sous `group_by` suivi de la **base de données** et **du ou des facteurs**, la précision des facteurs permet d'obtenir nos indices statistiques en fonction de leurs modalités.
- **Précision des statistiques et de la variable quantitative** : Sous `summarise` suivi des fonctions désirées, en l'occurrence le nombre d'observations (`n`), la moyenne (`mean`) et l'écart-type (`sd`) de la **variable quantitative**. Les noms `Nombre`, `Moyenne` et `EcartType` sont les noms que vous souhaitez attribuer aux colonnes pour la fonction désirée et peuvent être modifiés selon besoins.

Exemple 1: Nous souhaitons obtenir un tableau, décrivant le nombre d'observations, la moyenne et l'écart-type, du nombre de fautes attribuées ([Fautes](#)) en fonction de l'ethnie de l'agresseur ([Agresseur](#)) et de la victime ([Victime](#)) tirées de la base de données [Database](#).

```
group_by(Database, Agresseur, Victime) %>%
  summarise(
    Nombre = n(),
    Moyenne = mean(Fautes, na.rm = TRUE),
    EcartType = sd(Fautes, na.rm = TRUE)
  )
```

```
# A tibble: 4 x 5
# Groups:   Agresseur [2]
  Agresseur Victime Nombre Moyenne EcartType
  <fct>     <fct>   <int>   <dbl>      <dbl>
1 Noir       Noir      82     0.697     0.135
2 Noir       Blanc     82     0.599     0.152
3 Blanc      Noir      82     0.603     0.156
4 Blanc      Blanc     82     0.743     0.160
```

Exemple 2: Nous souhaitons obtenir un tableau, décrivant le nombre d'observations, la médiane et l'écart interquartile, du nombre de fautes attribuées ([Fautes](#)) en fonction de l'ethnie de l'agresseur ([Agresseur](#)) et de la victime ([Victime](#)), et du groupe ([Groupe](#)) auxquels les participant·es étaient affecté·es, variables tirées de la base de données [Database](#).

```
group_by(Database, Groupe, Agresseur, Victime) %>%
  summarise(
    Nombre = n(),
    Mediane = median(Fautes, na.rm = TRUE),
    EcartIQ = IQR(Fautes, na.rm = TRUE)
  )
```

```
# A tibble: 12 x 6
# Groups:   grp, Agresseur [6]
  grp        Agresseur Victime Nombre Mediane EcartIQ
  <fct>     <fct>     <fct>   <int>   <dbl>      <dbl>
1 Footballeurs Noir      Noir      43     0.688     0.188
2 Footballeurs Noir      Blanc     43     0.538     0.154
3 Footballeurs Blanc     Noir      43     0.5      0.143
4 Footballeurs Blanc     Blanc     43     0.7      0.3
5 Arbitres      Noir      Noir      17     0.812     0.125
6 Arbitres      Noir      Blanc     17     0.692     0.231
7 Arbitres      Blanc     Noir      17     0.786     0.357
8 Arbitres      Blanc     Blanc     17     0.8      0.2
9 Fans         Noir      Noir      22     0.688     0.0625
10 Fans        Noir      Blanc     22     0.692     0.288
11 Fans         Blanc     Noir      22     0.571     0.125
12 Fans         Blanc     Blanc     22     0.7      0.25
```

**Attention :** Lorsque vous avez transformé votre base de données pour effectuer une ANOVA à mesures répétées et que vous souhaitez les statistiques descriptives en fonction uniquement du facteur intersujet, alors vous devrez diviser le paramètre `n()` par le nombre de modalités du ou des facteur(s) intrasujet(s). Dans l'exemple ci-dessus, si vous souhaitez les statistiques descriptives de [Fautes](#) uniquement en fonction du [Groupe](#), alors précisez `n() / 4`, car les 4 modalités des facteurs intrasujets ne sont pas incluses. Cette procédure est nécessaire car vos participant·es se trouvent sur 4 lignes chacun·e depuis cette transformation (en somme, vous évitez les doublons)

#### 4.3.11 Test de la normalité (Kolmogorov-Smirnov) : lillie.test {nortest}

Le test de Lilliefors est une adaptation du test de Kolmogorov-Smirnov et permet de tester l'hypothèse nulle d'une distribution normale lorsque nous ne connaissons ni  $\mu$  ni  $\sigma$ . Il teste l'hypothèse nulle ( $H_0$ ) selon laquelle la distribution de la population dont provient l'échantillon est normalement distribuée. Donc, si  $p > .05$ , la distribution provient vraisemblablement d'une population normalement distribuée, et si  $p < .05$ , ce n'est pas le cas. Le résultat de ce test est à prendre avec précaution lorsque les observations sont plus élevées que  $n = 50$  : le test devient (trop) puissant et rapidement significatif (si  $n > 50$  et si  $p < .05$ , nous devons nous remettre à l'examen du boxplot ou de l'histogramme).

```
lillie.test(Database$IntellSL) ; length(Database$IntellSL)
```

```
> lillie.test(IntellSL) ; length(IntellSL)

Lilliefors (Kolmogorov-Smirnov) normality test

data: IntellSL
D = 0.085068, p-value = 0.06357
[1] 103
```

**Notation :** « La distribution de la variable *IntellSL* semble être tirée d'une population normalement distribuée,  $D(103) = .09$ ,  $p > .05$ . »

Dans le test de Lilliefors, le degré de liberté (ici de 103) est égal au nombre de valeurs de la variable. Si vous lancez la fonction `length`, vous obtiendrez directement sa valeur.

**Remarque :** Pour obtenir les tests de Kolmogorov-Smirnov en fonction d'un facteur (p. ex.: en fonction du genre des participant·es, ici notée `GENRE`), suivez la commande suivante.

**Attention :** Il est nécessaire que la variable quantitative et le facteur soient inclus dans la même base de données, en l'occurrence `Database`, raison pour laquelle elle est notée à deux reprises.

```
lillie.test(Database$IntellSL[Database$GENRE=="Femme"])
lillie.test(Database$IntellSL[Database$GENRE=="Homme"])
```

**Lorsque vous souhaitez connaître le nombre de participant·es par groupe :** Vous pouvez spécifier pour quelles lignes seront calculées les longueurs. En l'occurrence, nous souhaitons la `length` (ou *nombre d'observations*) de la variable quantitative `IntellSL` uniquement pour le genre féminin dans un premier temps, puis uniquement pour le genre masculin.

```
length(Database$IntellSL[Database$GENRE=="Femme"])
length(Database$IntellSL[Database$GENRE=="Homme"])
```

#### 4.3.12 Valeurs extrêmes et aberrantes (ou Outliers)

Déetecter les valeurs extrêmes et aberrantes permet de connaître les valeurs en-dehors des moustaches du boxplot. Ces valeurs déforment la normalité de la distribution de la variable. Cela est particulièrement utile lorsque le test de Kolmogorov-Smirnov est significatif ou lorsque l'histogramme ou le boxplot présentent une asymétrie inadéquate.

**Remarque :** La fonction ci-dessous est très pratique, mais n'a pas été trouvée sous forme de package. C'est pourquoi, il vous faut l'insérer telle quelle dans le script, puis la compiler entièrement. Elle ne permet d'effacer les valeurs extrêmes uniquement par variable quantitative indépendamment de facteurs. Si vous souhaitez supprimer des valeurs extrêmes apparaissant dans les boxplots en fonction de facteurs (dans l'ANOVA), alors utilisez la fonction `edit()`.

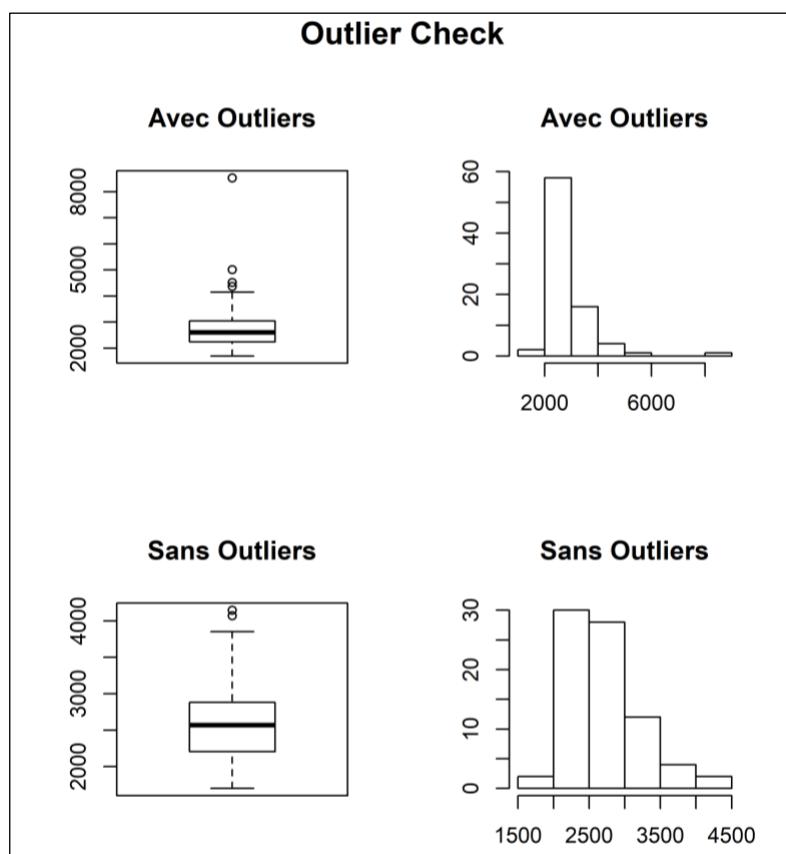
```
outlierKD <- function(dt, var) {
  var_name <- eval(substitute(var), eval(dt))
  tot <- sum(!is.na(var_name))
  na1 <- sum(is.na(var_name))
  m1 <- mean(var_name, na.rm = T)
  sd1 <- sd(var_name, na.rm = T)
  par(mfrow=c(2, 2), oma=c(0,0,3,0))
  boxplot(var_name, main="Avec Outliers", ylab=NA)
  hist(var_name, main="Avec Outliers", xlab=NA, ylab=NA)
  outlier <- boxplot.stats(var_name)$out
  mo <- mean(outlier)
  sdo <- sd(outlier)
  var_name <- ifelse(var_name %in% outlier, NA, var_name)
  boxplot(var_name, main="Sans Outliers", ylab=NA)
  hist(var_name, main="Sans Outliers", xlab=NA, ylab=NA)
  par(mfrow=c(1, 1))
  title("Outlier Check", outer=TRUE)
  na2 <- sum(is.na(var_name))
  message("Outliers identifiés: ", na2 - na1, " sur ", tot, " observations")
  message("Proportion d'Outliers: ", round((na2 - na1) / tot*100), digits=2), "%")
  message("Moyenne (Ecart-type) des Outliers: ",
         round(mo, digits=2), " (", round(sdo, digits=2), ")")
  m2 <- mean(var_name, na.rm = T)
  sd2 <- sd(var_name, na.rm = T)
  message("Moyenne (Ecart-type) de la mesure avec les Outliers: ",
         round(m1, digits=2), " (", round(sd1, digits=2), ")")
  message("Moyenne (Ecart-type) de la mesure sans les Outliers: ",
         round(m2, digits=2), " (", round(sd2, digits=2), ")")
  response <- readline(prompt="Souhaitez-vous retirer les Outliers et les
remplacer par NA ? [oui/non]: ")
  if(response == "y" | response == "yes" | response == "oui" | response=="Oui"){
    dt[as.character(substitute(var))] <- invisible(var_name)
    assign(as.character(as.list(match.call())$dt), dt, envir = .GlobalEnv)
    message("Retrait des Outliers accompli.", "\n")
    return(invisible(dt))} else{
    message("Aucun changement.", "\n")
    return(invisible(var_name))}}
# Selon https://datascienceplus.com/identify-describe-plot-and-removing-the-outliers-from-the-dataset/
# Adaptation et Traduction (FR)
```

Une fois la fonction créée, il est possible de procéder comme suit. Dans l'exemple, nous souhaitons savoir si la variable `t_b_b` tirée de la base de données `Database`, contient des valeurs extrêmes. Si elle en contient un nombre trop important, nous souhaitons les retirer. Respectez l'ordre présenté ci-dessous, soit en premier la base de données, puis la variable.

```
outlierKD(Database, t_b_b)
```

Cette fonction nous donne de nombreuses informations numériques, ainsi que les représentations graphiques (boxplots et histogrammes) **avec** et **sans** valeurs extrêmes.

```
> outlierKD(Database, t_b_b)
Outliers identifiés: 4 sur 82 observations
Proportion d'Outliers: 4.88%
Moyenne (Ecart-type) des Outliers: 5609.1 (1966.89)
Moyenne (Ecart-type) de la mesure avec les Outliers: 2795.8 (904.13)
Moyenne (Ecart-type) de la mesure sans les Outliers: 2651.53 (526.27)
Souhaitez-vous retirer les Outliers et les remplacer par NA ? [oui/non]: |
```



Dans la Console, il vous est demandé si vous souhaitez retirer (ou supprimer) ces valeurs extrêmes, ce à quoi vous pouvez directement répondre `oui` ou `non` à la suite du message. Si vous répondez `oui`, les valeurs extrêmes sont supprimées (et ainsi remplacées par des valeurs manquantes ou « `NA` »). Il est possible de le constater en demandant un descriptif de la variable. Cette suppression peut être particulièrement intéressante si elle est suivie d'un test de normalité de Kolmogorov-Smirnov afin de savoir si la variable est désormais normalement distribuée.

```
Souhaitez-vous retirer les Outliers et les remplacer par NA ? [oui/non]: oui
Retrait des Outliers accompli.

> Database$t_b_b
[1] 2848.143 2051.000 2549.286 2254.600 2552.600 4072.875 2499.500 1701.100 2698.000 3084.333 2267.500 2180.857 NA
[14] 2638.778 2062.167 3649.222 2701.250 NA 2773.667 NA 3286.400 2650.250 2022.000 3665.222 2737.286 3852.400
[27] 3321.167 3371.800 2695.200 2644.143 3081.111 3273.286 2525.700 2288.333 2145.889 3377.429 2551.111 2458.714 2322.429
[40] 2536.833 2716.333 2097.833 2597.200 2141.000 2205.800 3651.000 2424.556 2178.667 2041.833 2484.600 2135.571 2204.833
[53] 2099.100 2848.625 2298.625 2433.857 3041.200 3027.667 2613.143 3448.714 2943.250 3142.667 2716.375 3241.714 4145.200
[66] 2778.833 2315.000 2881.444 2673.667 2111.429 NA 2115.714 2060.800 2572.333 2526.667 1797.000 2811.625 2111.286
```

### 4.3.13 Graphiques

**Attention :** Dans les sous-chapitres suivants, lorsque vous insérez un facteur dans la commande, il est nécessaire de l'avoir préalablement factorisé et d'avoir spécifié le nom de ses modalités (cf. chapitres 4.2.7 et 4.2.8).

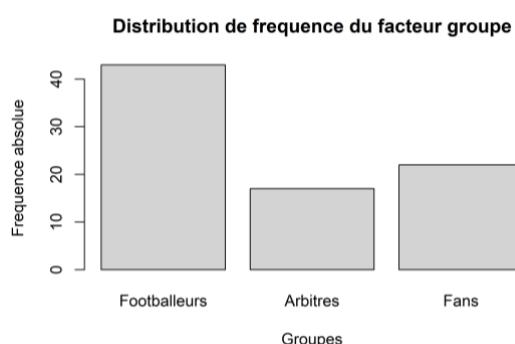
#### 4.3.13.1 Modifications des attributs d'un graphique

Vous pouvez modifier les attributs des graphiques obtenus par les fonctions `barplot`, `hist`, `plotNormalHistogram` et `boxplot` grâce à plusieurs paramètres dont voici les plus importants. Les paramètres décrits ci-dessous ne sont imputables qu'aux graphiques obtenus à partir du package `{graphics}`. Les autres graphiques, soit ceux obtenus grâce au package `{ggplot2}`, ont des paramètres sensiblement différents et ne seront pas traités dans ce sous-chapitre (mais peuvent être demandés grâce à la fonction `help`).

- **Couleur(s) des barres :** Sous `col=`, vous pouvez modifier la couleur des barres. Il est possible de n'indiquer qu'une seule couleur ou plusieurs suivant le nombre de barres : la commande varie légèrement entre une seule et plusieurs couleurs.
  - Pour une seule couleur : `col="lightblue"`.
  - Pour plusieurs couleurs : `col=c("white","grey","black")`
- **Couleur(s) des bordures de barres :** Sous `border=`, vous pouvez modifier la couleur de la bordure des barres. Tout comme la couleur des barres, il est possible de sélectionner plusieurs couleurs par le même procédé présenté précédemment.
- **Titre :** Sous `main=`, vous pouvez ajouter ou modifier le titre de votre graphique.
- **Nom des axes :** Sous `xlab=` pour l'axe des abscisses (*axe des x*) et `ylab=` pour l'axe des ordonnées (*axe des y*), vous pouvez ajouter ou modifier le nom des axes.
- **Etendue des axes :** Sous `xlim=c(...,...)` pour l'axe des abscisses (*axe des x*) et `ylim=c(...,...)` pour l'axe des ordonnées (*axe des y*), vous pouvez spécifier la valeur minimale et maximale, respectivement entre parenthèses, de l'étendue des axes.

Ci-dessous, un exemple de distribution de fréquences (`barplot`) de la variable `Groupe` tirée de la base de données `Database`, avec des barres en gris clair, une bordure de barres noire, un titre et des noms attribués aux axes.

```
barplot(table(Database$Groupe), col="lightgrey", border="black",
main = "Distribution de fréquence du facteur groupe",
xlab="Groupe", ylab="Fréquence absolue")
```



#### 4.3.13.2 Enregistrement d'un graphique en tant qu'image : `png {grDevices}`

Cette fonction permet d'enregistrer un graphique en tant qu'image (sous format \*.png) en qualité élevée. Plusieurs paramètres peuvent être modifiés comme décrits ci-dessous. Ceci est relativement utile pour les utilisateurs de Windows dont l'export d'images sous RStudio depuis l'onglet « Plots » est trop mauvaise pour être agréée par les normes de publication d'articles.

**Remarque :** Il est également possible d'enregistrer les graphiques sous d'autres formats disponibles dans le package `{grDevices}`, en l'occurrence `bmp`, `jpeg` et `tiff`. Dans ce chapitre, ne sera développé que le format `*.png`, format le plus approprié pour les graphiques et images contenant du texte dont la lisibilité est conservée.

- **Nom de fichier :** Directement accolé à la première parenthèse, entre guillemets, vous y attribuez le nom à l'image exportée en y précisant l'extension `.png`.
- **Largeur du graphique :** Sous `width=`, vous pouvez préciser la largeur du graphique en pouces (ou `inch`, en fonction de l'unité précisée).
- **Longueur du graphique :** Sous `height=`, vous pouvez préciser la longueur du graphique en pouces (ou `inch`, en fonction de l'unité précisée).
- **Unité :** Sous `units`, vous pouvez modifier l'unité de mesure des largeur et longueur de graphique, ici maintenue en `in` (soit `inch`).
- **Résolution :** Sous `res`, vous pouvez modifier la résolution de l'image, généralement maintenue à 600, offrant une résolution idéale pour un article.

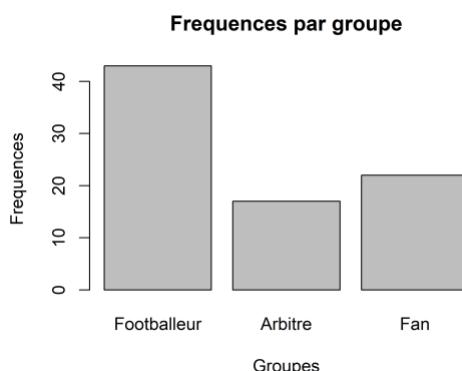
L'élément `dev.off()` doit être compilé afin de finaliser l'export d'images, sans quoi le fichier sera illisible. Cette commande doit être exécutée sur trois lignes distinctes et chacune compilée dans l'ordre respectif comme décrit ci-dessous. Une fois toutes compilées, votre image se trouvera dans votre dossier de projet.

```
png("Ex1Boxplot.png", width=7, height=5, units='in', res=800)
boxplot(Database$Age)
dev.off()
```

#### 4.3.13.3 Distribution de fréquences : `barplot {graphics}`

Les distributions de fréquences permettent une approche graphique des tableaux de fréquences décrits précédemment. En y, les fréquences absolues (ou  $n$ ) du facteur `Groupe`.

```
barplot(table(Database$Groupe))
```



**Remarque :** Si les modalités de votre variable nominale sont très nombreuses et/ou contiennent trop de caractères et que certains noms n'apparaissent pas, il est possible d'afficher les noms en vertical, de réduire les caractères et élargir les marges du graphique.

```
par(las=2) #Verticaliser les modalites de l'axe des x
par(mar=c(8,3,2,0)) #Adapter marge "c(bas, gauche, haut, droite)"
barplot(table(Database$Groupe), cex.names=.8) #Diminuer police
```

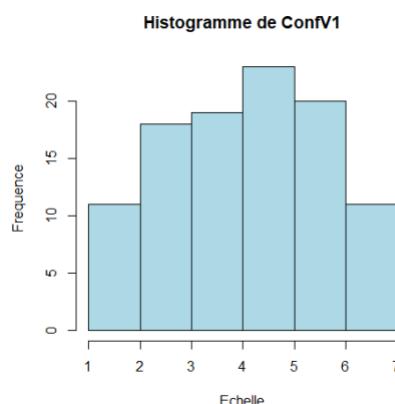
#### 4.3.13.4 Histogramme : `hist {graphics}`

Cette fonction est utilisée pour créer l'histogramme d'une variable donnée. Vous pouvez utiliser les mêmes paramètres que ceux décrits dans la modification des attributs de graphiques. A cela, s'ajoute un paramètre supplémentaire :

- **Étendue et incrément :** L'argument `breaks=seq(x, y, z)` permet de spécifier (x) la valeur minimale considérée par les barres, (y) la valeur maximale et (z) l'incrément. Attention à bien insérer les minimum et maximum de la variable (en cas d'incertitude, demander un descriptif complet de la variable avec les fonctions développées plus haut). Si vous spécifiez des limites qui négligent des données, vous risquez un message d'erreur. Si vous n'avez pas nécessité de spécifier une étendue ou incrément particulier, ce paramètre peut être omis de votre commande.

Exemple : En spécifiant `breaks=seq(1, 7, 0.5)`, vous obtiendrez des barres allant de 1 à 7 avec un incrément de 0.5 (ou une « largeur » de barre de 0.5 unité).

```
hist(Database$ConfV1, breaks=seq(1,7,1))
```



**Remarque :** Pour obtenir les histogrammes en fonction d'un facteur (p. ex.: en fonction du genre des participant·es), suivez la commande suivante. Dans cet exemple, nous souhaitons obtenir deux histogrammes, un pour la modalité Femme et un autre pour la modalité Homme du facteur `GENRE`.

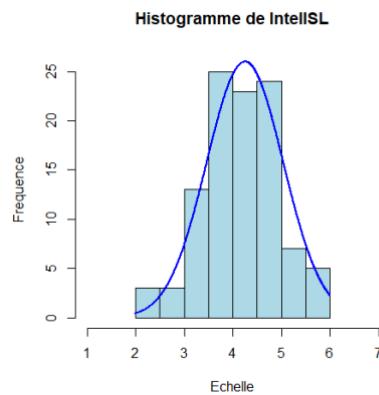
```
hist(Database$IntellSL[Database$GENRE=="Femme"])
hist(Database$IntellSL[Database$GENRE=="Homme"])
```

#### 4.3.13.5 Histogramme avec courbe normale : `plotNormalHistogram {rcompanion}`

**Remarque :** Le package peut être impossible à installer ou activer correctement. Si le cas se présente, compilez la commande `install.packages ("rcompanion", type="binary")`.

Cette fonction permet de superposer une courbe que devraient suivre les données si la variable était distribuée normalement. Cette option est très utile pour estimer la normalité des variables ainsi que pour détecter d'éventuelles valeurs extrêmes. Cette commande peut également comprendre le paramètre `breaks=seq(x, y, z)` développé dans le chapitre précédent. L'argument `prob=FALSE` maintient l'axe des y en fréquences absolues comme le fait la fonction `hist` développée plus haut.

```
plotNormalHistogram(Database$IntellSL, prob=FALSE)
```



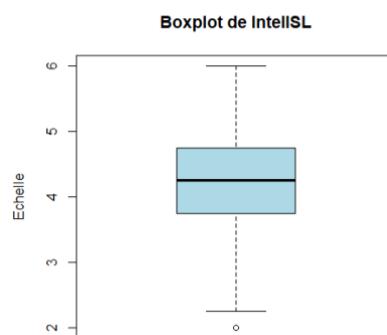
**Remarque :** Pour obtenir les histogrammes en fonction d'un facteur (p. ex.: en fonction du genre des participant·es), suivez la commande suivante. Dans cet exemple, nous souhaitons obtenir deux histogrammes, un pour la modalité Femme et un autre pour la modalité Homme de `GENRE`.

```
plotNormalHistogram(Database$IntellSL[Database$GENRE=="Femme"])
plotNormalHistogram(Database$IntellSL[Database$GENRE=="Homme"])
```

#### 4.3.13.6 Boxplot : `boxplot {graphics}, ggplot {ggplot2}`

Cette fonction permet de créer une boîte à moustaches avec une variable donnée. Tout comme la distribution de fréquences, il est tout à fait possible d'en modifier les attributs.

```
boxplot(Database$IntellSL)
```



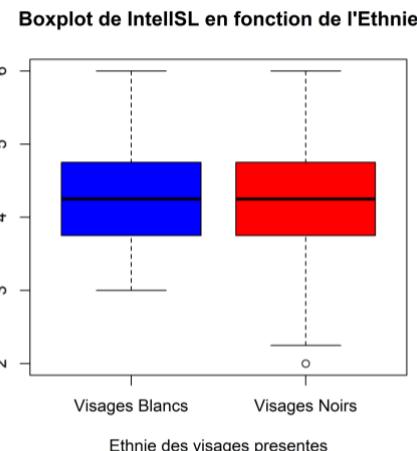
De plus, il est possible de demander **les valeurs descriptives du ou des boxplot·s**, soit [1,] la valeur de la moustache inférieure, [2,] du premier quartile (soit le « bas » de la boîte), [3,] de la médiane, [4,] du troisième quartile (soit le « haut » de la boîte) et [5,] la valeur de la moustache supérieure.

```
boxplot(Database$IntellSL)$stats
```

```
[,1]
[1,] 2.25
[2,] 3.75
[3,] 4.25
[4,] 4.75
[5,] 6.00
```

Situation 1 : Nous souhaitons créer un boxplot du jugement d'intelligence des visages sans lunettes en fonction de l'ethnie des visages présentés. Nous avons donc **1 variable quantitative** (`IntellSL`) et **1 variable nominale** (ou **facteur**) (`EthnieVisages`).

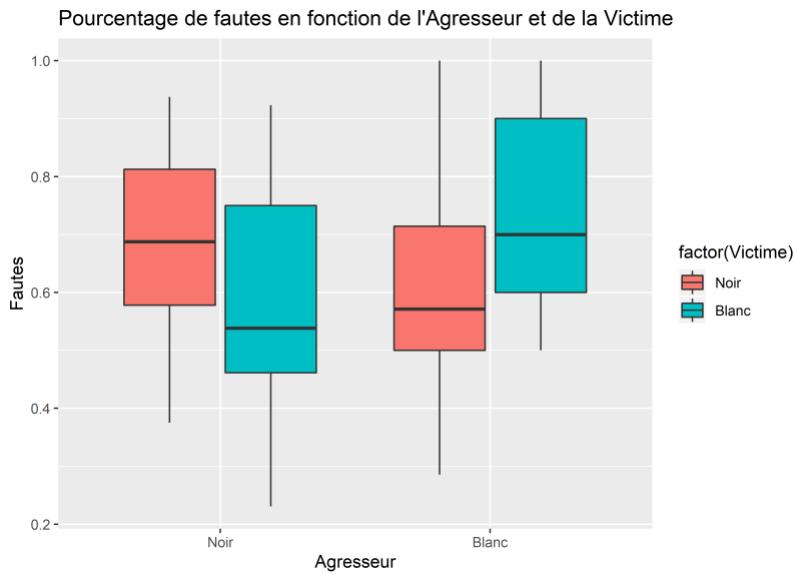
```
boxplot(Database$IntellSL ~ Database$EthnieVisages)
```



Situation 2 : Nous souhaitons créer un boxplot du nombre de fautes attribuées à la visualisation de courts extraits vidéo en fonction de l'ethnie de l'Agresseur (facteur 1, *peau noire* ou *peau blanche*) et de la Victime (facteur 2, *peau noire* ou *peau blanche*). Nous avons donc **1 variable quantitative** (`Fautes`) et **2 variables nominales** (`Agresseur` et `Victime`).

En utilisant la fonction `ggplot`, vous pouvez alors obtenir des boxplots réunis sous forme de **clusters**. Sous `fill=`, vous spécifiez le cluster (ce qui sera mis sous légende à droite de votre graphique). Sous `xlab` et `ylab`, vous pouvez spécifier le titre de l'axe désiré.

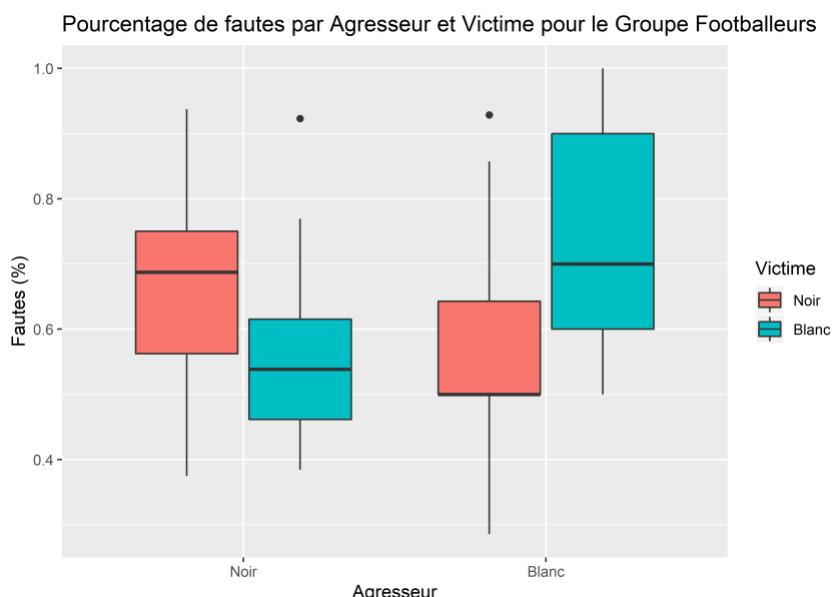
```
ggplot(Database, aes(Agresseur, Fautes, fill = Victime))+
  ggtitle("Pourcentage de fautes (...)") + ylab("Fautes (%)") +
  geom_boxplot()
```



Situation 3 : Nous souhaitons créer un boxplot du nombre de fautes attribuées à la visualisation de courts extraits vidéo en fonction de l'ethnie de l'Agresseur (facteur 1, *peau noire* ou *peau blanche*), de la Victime (facteur 2, *peau noire* ou *peau blanche*) et du groupe auquel ont été affecté·es les participant·es (facteur 3, *footballeurs*, *arbitres* ou *fans*). Nous avons donc **1 variable quantitative** (*Fautes*) et **3 variables nominales** (*Agresseur*, *Victime*, *Groupe*).

Lorsque vous avez plus de deux facteurs, il est nécessaire d'effectuer plusieurs graphiques. Pour cela, nous devons spécifier chaque modalité du troisième facteur (à choix) directement à la suite de la base de données, cela pour chacun des graphiques. En l'occurrence, nous souhaitons trois graphiques en fonction des trois modalités de la variable *Groupe*. Ci-dessous, un exemple pour la modalité *Footballeurs*.

```
ggplot(Database[Database$Groupe=="Footballeurs",], #Exemple Mod=1
       aes(Agresseur, Fautes, fill = Victime)) +
  ggtitle("Pourcentage de fautes (...)") + ylab("Fautes (%)") +
  geom_boxplot()
```



#### 4.3.13.6.1 Identifier les valeurs extrêmes d'un boxplot

Les valeurs extrêmes (et aberrantes) sont signalées par un cercle. Il est possible de demander le nombre (car ces cercles peuvent se superposer) et les valeurs de ces valeurs extrêmes. Dans cet exemple, nous souhaitons connaître le nombre de valeurs extrêmes ainsi que leur nombre de la variable `IntellSL`.

```
boxplot(Database$IntellSL)$out #Valeurs des VE  
length(boxplot(Database$IntellSL)$out) #Nombre de VE
```

#### 4.3.13.7 Diagramme en barres ou Graphique de moyennes

Fonctions et packages nécessaires : `ddply {plyr}` ; `ggplot {ggplot2}` ; `std.error {plotrix}`

Les graphiques de moyennes sont utiles pour représenter graphiquement les moyennes de variables quantitatives en fonction des modalités de facteurs. Différentes situations vous sont exposées, selon les variables de votre plan d'analyse.

Situation 1 : Nous souhaitons observer la moyenne et l'écart-type de l'âge en fonction du genre des participant·es sous forme graphique. Nous avons donc **1 variable quantitative** (`Age`) et **1 variable nominale ou facteur** (`Sexe`).

**Etape 1 :** Créer un tableau avec la moyenne et l'erreur-standard (ou l'écart-type (*SD*), à choix) de la variable quantitative (`Age`) en fonction des modalités du facteur (`Sexe`). Remarquez que la base de données est précisée (en l'occurrence `Database`) suivie du `facteur` et plus loin de la **variable quantitative**.

```
df <- ddply(Database, c("Sexe"), summarize,  
            Mean = mean(Age, na.rm=T),  
            SE    = std.error(Age, na.rm=T)) ; df
```

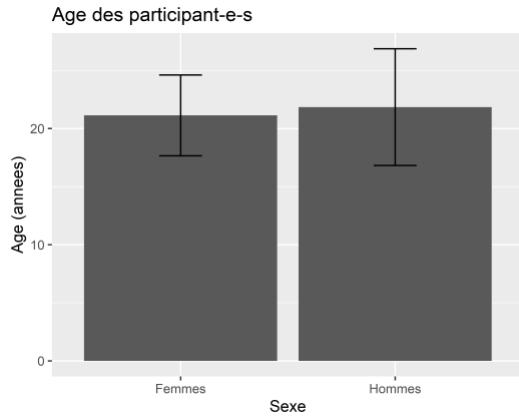
	Sexe	Mean	SD
1	Femmes	21.13514	3.477223
2	Hommes	21.84615	5.025474

**Remarque :** Dans le cas où un·e participant·e n'a aucune modalité attribuée dans le facteur, un tableau avec une ligne supplémentaire « NA » apparaît. Si tel est le cas, il suffit de retirer la ligne au numéro correspondant grâce à la commande suivante. Si cela n'est pas le cas, vous pouvez passer à l'étape suivante.

```
df <- df[-c(3),] ; df #Retrait des NA à la ligne 3
```

**Etape 2 :** Crédation finale du graphique. Les variable quantitative et facteur sont **en bleu** et vous permet de mieux vous situer. Sous `ggtitle`, vous précisez le titre à donner à votre diagramme en barres (à choix). Sous `ylab`, vous précisez le label de l'axe des y. Si vous souhaitez des barres d'erreur représentant l'intervalle de confiance (95%), remplacez les `SE` par `(1.96*SE)`.

```
ggplot(df, aes(x = Sexe, y = Mean)) +  
  geom_bar(stat = "identity", position = "dodge") +  
  ggtitle("Age des participant-e-s") + ylab("Age (annees)") +  
  geom_errorbar(aes(ymin = Mean-SE, ymax = Mean+SE), width = 0.2,  
                position = position_dodge(0.9))
```



Situation 2 : Nous souhaitons observer, sous forme graphique, la moyenne et l'écart-type de l'âge en fonction du genre des participant·es et de l'ethnie des visages présentés dans le questionnaire. Nous avons donc **1 variable quantitative** (`Age`) et **2 variables nominales** (`Sexe` et `EthnieVisages`). La procédure est identique pour les facteurs à plus de deux modalités. Notez que cette analyse (âge par groupes) permet de vérifier l'équivalence de l'âge de nos groupes pour l'expérience.

**Etape 1 :** Créez un tableau avec la moyenne et l'erreur-standard (ou l'écart-type (*SD*), à choix) de la variable quantitative (`Age`) en fonction des modalités des facteurs.

```
df <- ddply(Database, c("Sexe", "EthnieVisages"), summarize,
             Mean = mean(Age, na.rm=T),
             SE    = std.error(Age, na.rm=T)) ; df
```

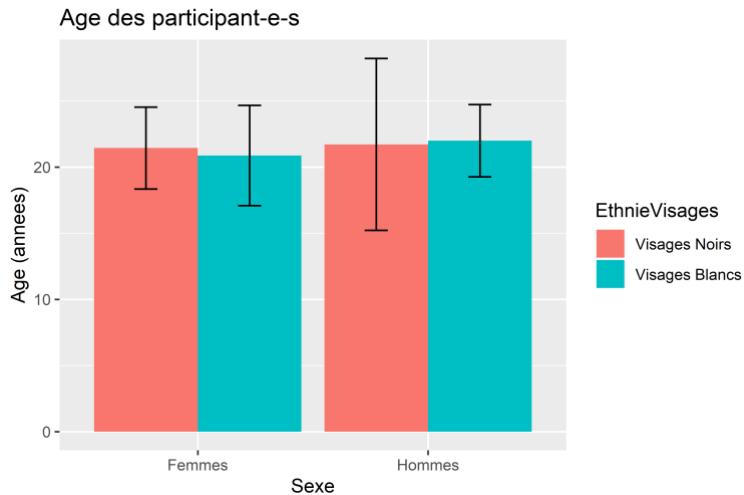
	Sexe	EthnieVisages	Mean	SD
1	Femmes	Visages Noirs	21.44118	3.096344
2	Femmes	Visages Blancs	20.87500	3.790592
3	Hommes	Visages Noirs	21.71429	6.497675
4	Hommes	Visages Blancs	22.00000	2.730301

**Remarque :** Dans le cas où un·e participant·e n'a aucune modalité attribuée dans le facteur, un tableau avec plusieurs lignes supplémentaires « NA » apparaissent. Si tel est le cas, il suffit de retirer les lignes au numéro correspondant grâce à la commande suivante. Si cela n'est pas le cas, vous pouvez passer à l'étape suivante.

```
df <- df[-c(5,6),] #Retrait des NA aux lignes 5 et 6
```

**Etape 2 :** Création finale du graphique. Sous `ggtitle`, vous précisez le titre à donner à votre diagramme en barres (à choix). Sous `ylab`, vous précisez le label de l'axe des y. Si vous souhaitez des barres d'erreur représentant l'intervalle de confiance (95%), remplacez les `SE` par `(1.96*SE)`. Sous `fill=`, vous spécifiez le cluster (ce qui sera mis sous légende à droite de votre graphique). Sous `xlab` et `ylab`, vous pouvez spécifier le titre de l'axe désiré.

```
ggplot(df, aes(x = Sexe, y = Mean, fill = EthnieVisages)) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Age des participant-e-s") + ylab("Age (annees)") +
  geom_errorbar(aes(ymin = Mean-SE, ymax = Mean+SE), width = 0.2,
                position = position_dodge(0.9))
```



Situation 3 : Nous souhaitons observer, sous forme graphique, la moyenne et l'écart-type du jugement d'intelligence de visages présentés avec lunettes et sans lunettes en fonction de l'ethnie du visage présenté. Nous avons donc **2 variables quantitatives** (l'intelligence des visages *avec* ou *sans* lunettes, par le facteur intrasujet appelé **Intelligence** et une variable quantitative **Score**) et **1 variable nominale** (**EthnieVisages**). Cette situation demande une manipulation supplémentaire, soit regrouper les deux variables quantitatives en un facteur.

**Attention :** Il vous faut transformer vos deux variables quantitatives en un seul facteur. Pour cela référez-vous au chapitre *Préparation pour 1 variable à mesures répétées* (4.4.4.3.1).

**Etape 1 :** Créez un tableau avec la moyenne et l'erreur-standard (ou l'écart-type (*SD*), à choix) de la variable quantitative (**Score**) en fonction des modalités des facteurs.

```
df <- ddply(Database, c("Intelligence", "EthnieVisages"), summarize,
             Mean = mean(Score, na.rm=T),
             SE    = std.error(Score, na.rm=T)) ; df
```

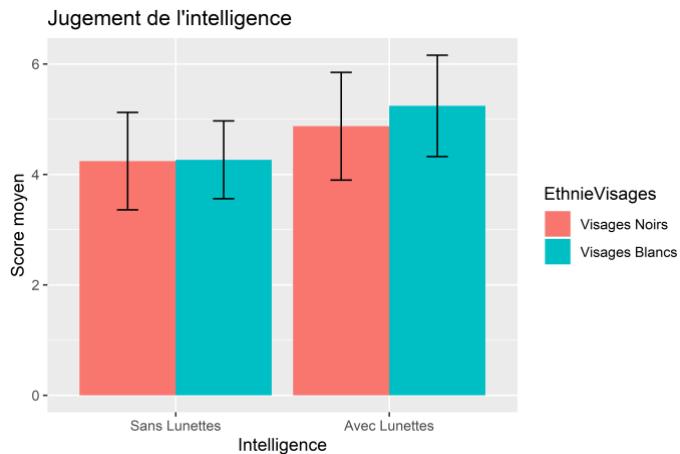
	Intelligence	EthnieVisages	Mean	SD
1	Sans Lunettes	Visages Noirs	4.239796	0.8823489
2	Sans Lunettes	Visages Blancs	4.263889	0.7044613
3	Avec Lunettes	Visages Noirs	4.872340	0.9764515
4	Avec Lunettes	Visages Blancs	5.240385	0.9181737

**Remarque :** Dans le cas où un·e participant·e n'a aucune modalité attribuée dans le facteur, un tableau avec plusieurs lignes supplémentaires « NA » apparaissent. Si tel est le cas, il suffit de retirer les lignes au numéro correspondant grâce à la commande suivante. Si cela n'est pas le cas, vous pouvez passer à l'étape suivante.

```
df <- df[-c(3,6),] #Retrait des NA aux lignes 3 et 6
```

**Etape 2 :** Création finale du graphique. Les variable quantitative et facteur sont **en bleu** et vous permet de mieux vous situer. Sous **ggtitle**, vous précisez le titre à donner à votre diagramme en barres (à choix). Sous **ylab**, vous précisez le label de l'axe des y. Si vous souhaitez des barres d'erreur représentant l'intervalle de confiance (95%), remplacez les *SD* par  $(1.96 \times SD)$ .

```
ggplot(df, aes(x = Lunettes, y = Mean, fill = EthnieVisages)) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Jugement de l'intelligence") + ylab("Score moyen") +
  geom_errorbar(aes(ymin = Mean-SE, ymax = Mean+SE), width = 0.2,
                position = position_dodge(0.9))
```



#### 4.3.13.8 Graphique d'interaction

Fonction et packages nécessaires : ddply, summarise {plyr} ; ggplot {ggplot2}

Les graphiques d'interaction sont utiles pour illustrer un effet d'interaction significatif.

Exemple : Nous souhaitons observer, sous forme graphique le score de jugement (`ScoreJugement`) donné à des visages en fonction du trait de jugement (`Traits`) du port ou non de lunettes (`Lunettes`), variables tirées de la base de données `Database`. Nous avons donc **1 variable quantitative et 2 variables nominales à respectivement 5 et 2 modalités**. Notons que l'effet d'interaction entre le port de lunettes et le trait de jugement s'est révélé significatif. Afin de faciliter l'interprétation de cet effet d'interaction comprenant dix moyennes différentes, nous optons ainsi pour un complément graphique aux comparaisons du test post-hoc de Tukey HSD.

**Etape 1 :** Créer un tableau (ici nommé arbitrairement `TraitLunINT`) avec la moyenne de la variable quantitative (`ScoreJugement`) en fonction des modalités des facteurs.

```
TraitLunINT <- ddply(Database, .(Traits, Lunettes),
                      summarise, val=mean(ScoreJugement, na.rm=T))
```

	Traits	Lunettes	val
1	Attrait	L	2.979773
2	Attrait	SL	3.519417
3	Conf	L	4.319579
4	Conf	SL	4.179612
5	Honnet	L	4.282362
6	Honnet	SL	4.019417
7	Intell	L	5.066343
8	Intell	SL	4.252427
9	Succes	L	4.964401
10	Succes	SL	4.087379

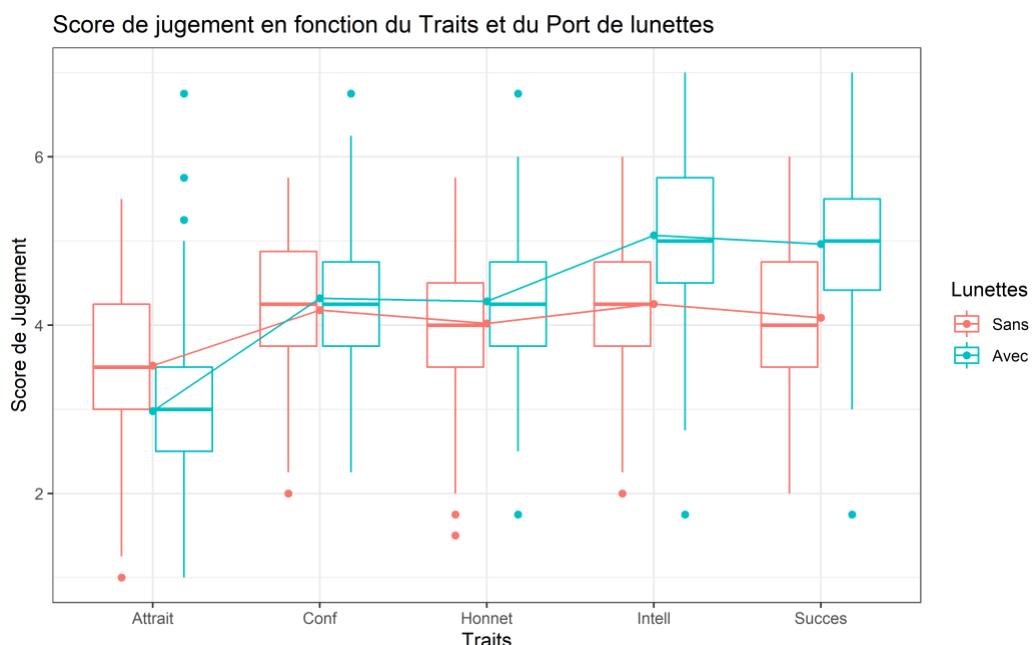
**Remarque :** Dans le cas où un·e participant·e n'a aucune modalité attribuée dans l'un des facteurs, un tableau avec plusieurs lignes supplémentaires « NA » apparaissent. **Si tel est le cas**, il suffit de retirer les lignes au numéro correspondant grâce à la commande suivante. Si cela n'est pas le cas, vous pouvez passer à l'étape suivante.

```
TraitLunINT <- TraitLunINT [-c(11,12),] #NA aux lignes 11 et 12
```

**Etape 2 :** Création finale du graphique. La commande suivante contient de nombreux éléments graphiques facultatifs :

- **Boîtes à moustaches :** Par `geom_boxplot()`, vous ajoutez des boîtes à moustaches (ou *boxplot*) au graphique d’interaction. Ces boîtes à moustaches seront accompagnées de points signalant les valeurs extrêmes.
- **Points de moyenne :** Par `geom_point()`, vous ajoutez un point signalant la moyenne de chacune des combinaisons. Ces points se superposeront à la droite de moyennes obtenue grâce à `geom_line()`. Notez que ces points peuvent ne pas être alignés à la médiane des boxplots (si divergence entre médianes et moyennes).
- **Droites de moyennes :** Par `geom_line()`, vous ajoutez une droite de moyennes. Cette droite permet d’observer les croisements entre modalités et de signaler les interactions.
- **Titre de graphique et label d’axe:** Par `ggtitle()` et `ylab()` respectivement, vous ajoutez un titre à votre graphique et un label à l’axe des ordonnées. Il est également possible d’ajouter un label à l’axe des abscisses sous `xlab()`.

```
ggplot(Database, aes(x=Traits, y=ScoreJugement, colour=Lunettes)) +
  geom_boxplot() +
  geom_point(data = TraitLunINT, aes(y = val)) +
  geom_line(data = TraitLunINT, aes(y = val, group = Lunettes)) +
  ggtitle("Score en fonction de...") + ylab("Score de Jugement") +
  theme_bw()
```



## 4.4 Fonctions de statistiques inférentielles

Dans le chapitre 4.3, il a été question des fonctions descriptives. Dans ce chapitre, il est question de tester des hypothèses statistiques. Avant de présenter les tests, nous proposons un tableau qui vous aidera à décider quelle analyse est la plus adéquate selon le type de variables (nominale, ordinaire ou quantitative) dont il est question dans votre base de données et selon le rôle de chacune de celles-ci (variable indépendante ou dépendante).

### 4.4.1 Tableau de décision statistique

VI VD VI	Variable(s) nominale(s) (p.ex.: genre, faculté)	Variable(s) ordinaire(s) (p.ex.: rang, échelle)	Variable(s) quantitative(s) (p.ex.: âge, temps de réponse)
<b>Variable nominale</b>	<b>Khi<sup>2</sup></b> (2 variables nominales)	<i>Tests non paramétriques :</i> <b>U de Mann-Whitney</b> (2 groupes indépendants) <b>Kruskal-Wallis</b> ( $n$ groupes indépendants) <b>Wilcoxon</b> (2 groupes appariés) <b>ANOVA de Friedman</b> ( $n$ groupes appariés)	<b>T-test</b> (comparaison de 2 moyennes) <b>(M)ANOVA</b> (comparaison de $n$ moyennes, d'une ou plusieurs VI, d'une ou plusieurs VD)
<b>Variable ordinale</b>		<i>Test non paramétrique :</i> <b>Corrélation (Spearman)</b> (2 variables ordinaires. Toutes les variables sont de même niveau, il n'y a pas de VI ni VD)	
<b>Variable(s) quantitatives(s)</b>	<b>Régression logistique</b> ( $n$ variables quantitatives et/ou nominales. VI = prédicteurs, variables nominales ou quantitatives VD = variable binomiale)		<b>Corrélation (Pearson)</b> (2 variables quantitatives. Toutes les variables sont de même niveau, il n'y a pas de VI ni VD) <b>Régression linéaire simple</b> (2 variables quantitatives. VI = prédicteur, VD = critère) <b>Régression lin. multiple</b> ( $n$ variables quantitatives. VIs = plusieurs prédicteurs, VD = critère)

#### 4.4.2 Khi-carré : chisq.test {stats}

Cette analyse teste l'existence d'un lien entre deux variables nominales. Elle permet de déterminer si la différence entre deux distributions de fréquence est attribuable au hasard ( $H_0$ ) ou si cette différence est trop importante pour être simplement expliquée par le hasard ( $H_1$ ).

Exemple : Nous souhaitons tester l'existence d'un lien entre le groupe d'âge (`Gr_age`, « 70-80 » ou « 81-100 ») et le genre des participant·es (`Genre`, « Femme » ou « Homme »). Ces deux variables sont tirées de la base de données `Database`.

- **Fréquences observées** : L'élément `$observed` permet de connaître les fréquences observées de la variable. L'élément accolé, en l'occurrence `CSqAgeGen`, est l'analyse du chi-carré effectuée dans la commande précédente.
- **Fréquences attendues** : L'élément `$expected` permet de connaître les fréquences attendues (devant être supérieures à 5 pour obtenir un résultat fiable) de la variable suivant la proportion souhaitée. L'élément accolé, en l'occurrence `CSqAgeGen`, est l'analyse du chi-carré effectuée dans la commande précédente.

```
CSqAgeGen <- chisq.test(Database$Genre, Database$Gr_age,  
                         correct=FALSE)  
CSqAgeGen ; CSqAgeGen$observed ; CSqAgeGen$expected
```

```
Pearson's Chi-squared test  
  
data: Genre and Gr_age  
X-squared = 2.3709, df = 1, p-value = 0.1236  
  
> CSqAgeGen$observed  
    Gr_age  
Genre   70-80 81-100  
  Femmes   13    16  
  Hommes   11     5  
> CSqAgeGen$expected  
    Gr_age  
Genre   70-80     81-100  
  Femmes 15.466667 13.533333  
  Hommes  8.533333  7.466667
```

Pour connaître la valeur des fréquences relatives (ou pourcentages) à insérer dans la notation finale, veuillez suivre le chapitre suivant *Tableaux croisés*.

#### 4.4.2.1 Tableaux croisés : CrossTable {gmodels}

Pour obtenir un tableau croisé des fréquences obtenues, procédez comme suit. Dans l'exemple, nous avons demandé un tableau croisé des deux variables que nous avons précédemment incluses dans l'analyse.

```
CrossTable(Database$Genre, Database$Gr_age)
```

Ce tableau croisé nous donne de nombreuses informations supplémentaires :

- **La première ligne** nous donne chacune de nos fréquences obtenues.
- **La deuxième ligne** nous donne la contribution au  $\chi^2$  de chacune de nos fréquences obtenues en rapport aux fréquences attendues respectives. Ceci est utile pour les grands tableaux de contingence, afin de savoir où sont les liens s'il y en a.

- **Les lignes suivantes** nous donnent les fréquences relatives (ou pourcentages) en fonction des lignes, des colonnes ou du total de la table. Ceci est intéressant pour la phrase de conclusion. Dans celle-ci, vous pouvez utiliser deux pourcentages, entourés de la même couleur dans le tableau ci-dessous (plusieurs combinaisons sont possibles). Dans la phrase de conclusion ci-dessous, la combinaison des pourcentages entourés en vert a été choisie.

Cell Contents																			
N																			
Chi-square contribution																			
N / Row Total																			
N / Col Total																			
N / Table Total																			
Total Observations in Table: 45																			
<table border="1"> <thead> <tr> <th>Genre</th> <th>Gr_age 70-80</th> <th>81-100</th> <th>Row Total</th> </tr> </thead> <tbody> <tr> <td>Femmes</td> <td>13 0.393 <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">0.448</span> 0.542 0.289</td> <td>16 0.450 <span style="border: 1px solid orange; border-radius: 50%; padding: 2px;">0.552</span> 0.762 0.356</td> <td>29 0.644</td> </tr> <tr> <td>Hommes</td> <td>11 0.713 <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">0.688</span> 0.458 0.244</td> <td>5 0.815 <span style="border: 1px solid orange; border-radius: 50%; padding: 2px;">0.312</span> 0.238 0.111</td> <td>16 0.356</td> </tr> <tr> <td>Column Total</td> <td>24 0.533</td> <td>21 0.467</td> <td>45</td> </tr> </tbody> </table>				Genre	Gr_age 70-80	81-100	Row Total	Femmes	13 0.393 <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">0.448</span> 0.542 0.289	16 0.450 <span style="border: 1px solid orange; border-radius: 50%; padding: 2px;">0.552</span> 0.762 0.356	29 0.644	Hommes	11 0.713 <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">0.688</span> 0.458 0.244	5 0.815 <span style="border: 1px solid orange; border-radius: 50%; padding: 2px;">0.312</span> 0.238 0.111	16 0.356	Column Total	24 0.533	21 0.467	45
Genre	Gr_age 70-80	81-100	Row Total																
Femmes	13 0.393 <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">0.448</span> 0.542 0.289	16 0.450 <span style="border: 1px solid orange; border-radius: 50%; padding: 2px;">0.552</span> 0.762 0.356	29 0.644																
Hommes	11 0.713 <span style="border: 1px solid red; border-radius: 50%; padding: 2px;">0.688</span> 0.458 0.244	5 0.815 <span style="border: 1px solid orange; border-radius: 50%; padding: 2px;">0.312</span> 0.238 0.111	16 0.356																
Column Total	24 0.533	21 0.467	45																

**Notation :** « Bien qu'il y ait moins de femmes dans le groupe d'âge allant de 70 à 80 ans (54% de femmes) que dans le groupe de 81 à 100 ans (76% de femmes), la différence de proportions n'est pas significative,  $\chi^2(1) = 2.37, p > .05$ . »

#### 4.4.2.2 Khi-carré d'ajustement : une variable nominale

Le Chi-carré d'ajustement permet de vérifier si la proportion des fréquences des modalités d'une variable nominale sont égales à une proportion théorique donnée. Cette commande demande de transformer la variable nominale en table de fréquences et d'en préciser la proportion théorique testée.

- **Proportion théorique testée :** Sous `p=c(x, 1-x)=`, vous pouvez préciser la proportion théorique. Deux manières d'écrire cette proportion sont possibles, soit la forme factorielle (p. ex.: 1/2 ou 5/10), soit la forme décimale (p. ex.: 0.5).

Exemple 1 : Nous souhaitons vérifier si la proportion de femmes et d'hommes est égale à une proportion **50:50**.

```
CSqGr_age <- chisq.test(table(Database$Gr_age), p=c(0.5,0.5))
CSqGr_age ; CSqGr_age$observed ; CSqGr_age$expected
CrossTable(Database$Gr_age) #Tableau croise Gr_age (pourcentages)
```

```

Chi-squared test for given probabilities

data: table(Gr_age)
X-squared = 0.2, df = 1, p-value = 0.6547

> CSqGr_age$observed
Gr_age
70-80 81-100
24      21
> CSqGr_age$expected
70-80 81-100
22.5   22.5

```

**Notation :** La proportion d'individus de la tranche d'âge 70-80 ans et d'individus de la tranche d'âge 81-100 ans n'est pas significativement différente (53% de 70-80 ans et 47% de 81-100 ans),  $\chi^2(1) = .20, p > .05$ .

Exemple 2 : Nous souhaitons vérifier si la proportion de femmes et d'hommes est égale à une proportion 2/3 : 1/3.

```

CSqGenre21.3 <- chisq.test(table(Database$Genre), p=c(2/3,1/3))
CSqGenre21.3 ; CSqGenre21.3$observed ; CSqGenre21.3$expected
CrossTable(Database$Genre) #Tableau croise Genre (pourcentages)

```

```

Chi-squared test for given probabilities

data: table(Genre)
X-squared = 0.1, df = 1, p-value = 0.7518

> CSqGenre21.3$observed
Genre
Femmes Hommes
29      16
> CSqGenre21.3$expected
Femmes Hommes
30      15

```

**Notation :** La proportion de femmes et d'hommes n'est pas significativement différente d'une proportion 1/3 : 2/3 (64% de femmes et 36% d'hommes),  $\chi^2(1) = .10, p > .05$ .

#### 4.4.3 T-test : t.test {stats}

Le test du T de Student (ou t-test) permet de tester l'existence de différences entre moyennes de variables quantitatives. Plusieurs cas sont possibles : le **t-test à échantillon unique** (moyenne empirique vs. moyenne théorique), le **t-test à échantillons indépendants** (moyennes provenant de deux groupes indépendants constitués par une variable nominale) et le **t-test à échantillons appariés** (moyennes provenant de deux groupes appariés). Plusieurs paramètres sont à prendre en compte dans la commande d'analyse du t-test.

- **Moyenne théorique :** *Présente uniquement pour le t-test à échantillon unique.* Le paramètre `mu=` permet de spécifier la valeur de la moyenne théorique.
- **Hypothèse alternative :** Le paramètre `alternative=` permet de spécifier l'hypothèse alternative. Voici les possibilités :
  - **Hypothèse bilatérale** (`alternative="two.sided"`) : Les deux moyennes des deux variables ne sont pas égales (ou la différence entre les deux moyennes est différente de 0).

- **Hypothèse unilatérale 1** (`alternative="greater"`) : La moyenne de la première variable insérée est plus grande que la moyenne de la seconde (ou la différence entre les deux moyennes est supérieure à 0).
- **Hypothèse unilatérale 2** (`alternative="less"`) : La moyenne de la première variable insérée est plus petite que la moyenne de la seconde (ou la différence entre les deux moyennes est inférieure à 0).
- **Egalité des variances** : Le paramètre `var.equal=` est un paramètre logique et vous permet de préciser si les variances peuvent être considérées égales ou non en fonction des résultats obtenus par le test de Levene (cf. ci-dessous).

#### 4.4.3.1 T-test à échantillon unique (one sample t-test)

Le t-test à échantillon unique permet de tester l'existence d'une différence significative entre la moyenne d'une variable quantitative (ou moyenne empirique) et une moyenne théorique (fixée d'avance). La moyenne théorique peut être, entre autres, le milieu d'une échelle de Likert, un temps donné ou la valeur moyenne du QI (100).

Exemple : Nous souhaitons savoir si la moyenne de notre variable `RespTime`, variable tirée de la base de données `Database`, est significativement inférieure à une valeur donnée  $\mu$  (mu) de 2'000ms.

```
t.test(Database$RespTime, alternative="less", mu=2000)
```

```
One Sample t-test

data: RespTime
t = -2.5799, df = 44, p-value = 0.006649
alternative hypothesis: true mean is less than 2000
95 percent confidence interval:
-Inf 1929.303
sample estimates:
mean of x
1797.272
```

Notez que la fonction vous permet de connaître la moyenne, mais également l'hypothèse alternative en lien avec ce que vous avez inséré après `alternative` (ce qui permet de vérifier que l'hypothèse alternative a correctement été spécifiée).

**Notation** : « La moyenne du temps de réponse ( $M = 1797.27$ ) est significativement inférieure à la valeur de 2'000ms,  $t(44) = -2.58, p < .05$ . »

#### 4.4.3.2 T-test à échantillons (ou groupes) indépendants (independent samples t-test)

Fonction et package nécessaires : `leveneTest {car}, {data.table}`

Le t-test à échantillons indépendants permet de tester l'existence d'une différence significative entre deux moyennes d'une VD (provenant de deux groupes distincts formés par une variable nominale ou facteur, soit la VI).

Exemple : Nous souhaitons savoir si l'honnêteté des visages sans lunettes `HonestSL` est jugée différemment selon l'ethnie du visage présenté (`EthnieVisages`), variables tirées de la base de données `Database`. Précisons que chaque participant·e n'a eu à juger des visages que d'une seule couleur de peau, soit *noire*, soit *blanche* (d'où échantillons indépendants).

**Attention :** Pour que les commandes suivantes soient exécutées correctement, il est nécessaire que la VI soit bel et bien considérée comme un facteur par R (cf. chapitre 4.2.7). De plus, afin de faciliter la lecture des tableaux, renommez les modalités de votre facteur avec le label approprié. Voici un exemple avec la VI `EthnieVisages`. Cette commande est développée en détails au chapitre *Renommer les modalités d'une variable nominale* (4.2.8).

```
Database$EthnieVisages <- factor(Database$EthnieVisages,
levels=c(1,2), labels=c("Visages Noirs","Visages Blancs"))
```

**Tout d'abord**, il est nécessaire de vérifier l'homogénéité des variances grâce au test de Levene, qui est l'une des conditions d'application du t-test à échantillons indépendants. Ci-dessous, les résultats nous indiquent que les deux groupes n'ont pas une variance homogène,  $F(1, 101) = 5.28, p < .05$ .

```
leveneTest(HonnetSL~EthnieVisages, data=Database, center=mean)
```

```
Levene's Test for Homogeneity of Variance (center = median)
  Df F value Pr(>F)
group  1  5.2776 0.02367 *
      101
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

**Dans le cas d'une non homogénéité des variances**, lorsque vous lancez le t-test, il vous faut préciser que les variances ne sont pas égales :

```
t.test(HonnetSL~EthnieVisages, data=Database,
       alternative="less", var.equal=FALSE)
```

**Dans le cas d'une homogénéité des variances**, lorsque vous lancez le t-test, il vous faut préciser que les variances sont égales :

```
t.test(HonnetSL~EthnieVisages, data=Database,
       alternative="less", var.equal=TRUE)
```

```
Welch Two Sample t-test

data: HonnetsL by EthnieVisages
t = 0.11991, df = 83.144, p-value = 0.5476
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
 -Inf 0.3175552
sample estimates:
mean in group 1 mean in group 2
 4.030612     4.009259
```

**Rappel :** Pour obtenir les moyennes et écart-types en fonction des modalités, référez-vous au chapitre *Statistiques descriptives* (4.3.10). Voici un exemple avec la variable quantitative `HonnetSL` et le facteur `EthnieVisage`.

```
group_by(Database, EthnieVisages) %>%
  summarise(
    Nombre = n(),
    Moyenne = mean(HonnetSL, na.rm = TRUE),
    EcartType = sd(HonnetSL, na.rm = TRUE)
  )
```

EthnieVisages	Nombre	Moyenne	EcartType
	<int>	<dbl>	<dbl>
Visages Noirs	49	4.03	1.05
Visages Blancs	54	4.01	0.709

Lorsque vous précisez l'hypothèse alternative, dans le cas d'un t-test à échantillons indépendants, vérifiez l'ordre dans lequel a été codée votre VI (ici, les visages de peau noire sont la modalité 1 et les visages de peau blanche la modalité 2 : ainsi, nous formulons l'hypothèse que la moyenne du jugement d'honnêteté des visages de peau noire sans lunettes est moins élevée que la moyenne des visages de peau blanche sans lunettes (soit une supposition de discrimination).

**Notation :** « Les participant·es n'ont pas jugé les visages de peau noire sans lunettes ( $M = 4.03$ ,  $SD = 1.05$ ) comme moins honnêtes que les visages de peau blanche sans lunettes ( $M = 4.01$ ,  $SD = .71$ ),  $t(83.14) = .12$ ,  $p > .05$ . »

#### 4.4.3.3 T-test à échantillons appariés ou à mesures répétées (paired samples t-test)

Le t-test à échantillons appariés (ou mesures répétées) permet de tester l'existence d'une différence significative entre deux moyennes mesurées chez les mêmes sujets (ou sujets appariés par un critère extérieur, p. ex. membres du même couple). Dans cette analyse, on crée une VI à mesure répétées à partir de deux variables quantitatives (dans l'exemple ci-dessous, à partir des deux variables quantitatives "jugement de l'honnêteté des visages sans lunettes" et "jugement de l'honnêteté des visages avec lunettes", on crée une VI = présence/absence de lunettes, la VD devient le jugement de l'honnêteté).

Exemple : Nous souhaitons savoir si l'honnêteté des visages avec lunettes ([HonnetL](#)) et sans lunettes ([HonnetSL](#)) est jugée différemment par les mêmes participant·es, variables tirées de la base de données [Database](#). Précisons que les participant·es ont à la fois jugé des visages *avec* lunettes et *sans* lunettes (d'où la notion de mesures répétées).

```
t.test(HonnetSL, HonnetL, data=Database,
       alternative="two.sided", paired=TRUE)
```

```
Paired t-test

data: HonnetSL and HonnetL
t = -2.8363, df = 99, p-value = 0.005535
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.44613755 -0.07886245
sample estimates:
mean of the differences
-0.2625
```

**Notation :** « Les participant·es ont jugé les visages portant des lunettes ( $M = 4.28$ ,  $SD = .89$ ) différemment des visages ne portant pas de lunettes ( $M = 4.02$ ,  $SD = .88$ ),  $t(99) = -2.84$ ,  $p < .05$ . »

#### 4.4.4 ANOVA : aov {stats}

L'ANOVA permet de tester l'existence de différences entre moyennes d'une ou plusieurs variables(s) quantitative(s) (ou VDs), en fonction d'une ou plusieurs variable(s) nominale(s) (ou facteurs, VIs), qui peuvent comporter plus de deux modalités, contrairement au t-test. La fonction de ce test est de créer un modèle d'analyse de variance.

- ~ : Ce caractère sépare les variables dépendantes des variables indépendantes.
- \* : Ce caractère sépare les variables indépendantes tout en étudiant leur effet d'interaction.
- + : Ce caractère sépare les variables indépendantes sans étudier leur effet d'interaction.

##### 4.4.4.1 ANOVA à un facteur intersujet (One-way ANOVA)

Fonctions et packages nécessaires : `leveneTest {car}` ; `TukeyHSD {stats}` ; `etaSquared {lsr}`

L'ANOVA à un facteur intersujet permet de tester l'existence d'une différence significative entre plusieurs (2 ou plus) moyennes d'une variable quantitative (VD) provenant de plusieurs groupes formés par une variable nominale (VI ou facteur).

Exemple : Nous souhaitons savoir si *trois* groupes de participant·es (`Groupe`), soit des *joueurs*, des *arbitres* et des *fans*, ont des scores moyens de racisme (`ScoreRacisme`) différents, variables tirées de la base de données `Database`. Nous avons donc un facteur à 3 modalités (VI) et une variable quantitative étant le score moyen de racisme (VD).

**Attention** : Pour que les commandes suivantes soient exécutées correctement, il est nécessaire que la VI soit bel et bien considérée comme un facteur par R (cf. chapitre 4.2.7). De plus, afin de faciliter la lecture des tableaux, renommez les modalités de votre facteur avec le label approprié (cf. chapitre 4.2.8).

Dans les conditions d'application du test, l'**homogénéité des variances** de nos différents groupes doit être vérifiée grâce au **test de Levene**.

```
leveneTest(ScoreRacisme~Groupe, data=Database, center=mean)
```

```
Levene's Test for Homogeneity of Variance (center = mean)
  Df F value Pr(>F)
group  2  0.0318 0.9687
      75
```

Après cela, vous devez créer une nouvelle variable qui est l'**ANOVA** pour finalement en demander un résumé : cela vous permettra d'avoir les informations nécessaires pour savoir s'il existe ou non un effet de votre VI.

```
ANOVA_RbyG <- aov(ScoreRacisme~Groupe, data=Database)
summary(ANOVA_RbyG)
```

**Rappel** : Pour obtenir les moyennes et écart-types en fonction des modalités, référez-vous au chapitre *Statistiques descriptives* (4.3.10).

```
        Df Sum Sq Mean Sq F value    Pr(>F)
Group      2   3.838   1.9188   6.609  0.00227 ***
Residuals  75 21.776   0.2903
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
4 observations deleted due to missingness
```

# A tibble: 3 x 4	groupe	Nombre	Moyenne	EcartType
	<fct>	<int>	<dbl>	<dbl>
1	Footballeurs	43	2.47	0.553
2	Arbitres	17	2.66	0.556
3	Fans	22	2.05	0.498

**Remarque :** Nous savons *seulement* qu'il existe un effet (différence entre les moyennes des groupes, mais ne savons pas exactement où se situent les différences. En effet, nous avons trois groupes : il se pourrait qu'ils soient tous différents, mais également qu'un seul soit différent des autres, etc. Pour savoir où se situent ces différences, vous devez utiliser **un test post-hoc**, comme le test **HSD de Tukey** : Dans le cas où l'ANOVA n'indique pas d'effet significatif, ou s'il n'y a que deux groupes, il n'est pas nécessaire de poursuivre avec un test de Tukey HSD.

#### TukeyHSD (ANOVA\_RbyG)

Tukey multiple comparisons of means 95% family-wise confidence level				
Fit: aov(formula = ScoreRacisme ~ groupe, data = Database)				
\$groupe	diff	lwr	upr	p adj
Arbitres-Footballeurs	0.1944444	-0.2031707	0.59205957	0.4750918
Fans-Footballeurs	-0.4181097	-0.7571970	-0.07902235	0.0117216
Fans-Arbitres	-0.6125541	-1.0530415	-0.17206677	0.0038733

Grâce au test post-hoc, vous pouvez savoir où se situent les différences significatives de votre effet. En l'occurrence, deux différences (sur les trois possibles) sont significatives, soit entre les fans et footballeurs, et entre les fans et arbitres.

Un effet significatif peut être détecté grâce à des échantillons très importants, mais sans grands écarts entre les moyennes. La **taille d'effet** renseigne sur la grandeur des effets significatifs observés. Les tailles d'effets communément utilisées dans l'analyse de variance sont le  $\eta^2$  (éta-carré) et le  $\eta^2_{partiel}$ <sup>1</sup>. Elles font référence à la force ou à la magnitude de l'association, soit la proportion de la variance totale de la VD expliquée par les modalités de chaque VI.

#### etaSquared (ANOVA\_RbyG)

```
> etaSquared(ANOVA_RbyG)
  eta.sq eta.sq.part
Group 0.149826    0.149826
```

**Notation :** « Il existe un effet principal du groupe sur le score moyen de racisme,  $F(2, 75) = 6.61$ ,  $p < .05$ ,  $\eta_p^2 = .15$ . Un test de comparaisons post-hoc de Tukey HSD (avec un seuil de significativité fixé à  $p = .05$ ) indique que le score moyen de racisme des arbitres ( $M = 2.66$ ,  $SD = .56$ ) et des footballeurs ( $M = 2.47$ ,  $SD = .55$ ) ne diffèrent pas, mais sont significativement différents de celui des fans ( $M = 2.05$ ,  $SD = .50$ ). »

<sup>1</sup> Les tailles d'effet données par les  $\eta^2$  et  $\eta^2_{partiel}$  ne sont pas à confondre (elles sont égales pour l'ANOVA à 1 facteur, mais pas dès qu'il y en a plus qu'un). La valeur donnée par  $\eta^2$  représente la proportion de la variabilité totale de la variable dépendante expliquée par la variable indépendante. Il est le rapport de la somme des carrés de l'effet (pour chaque modalité) sur la somme des carrés totale (soit  $\eta^2 = SS_{Effect}/SS_{Total}$ ). La valeur donnée par  $\eta^2_{partiel}$  est plus complexe, moins recommandée dans la recherche scientifique et ajoute la somme des carrés résiduels au dénominateur (soit  $\eta^2_{partiel} = SS_{Effect}/(SS_{Effect} + SS_{Error})$ ).

#### 4.4.4.2 ANOVA à plusieurs facteurs intersujets (Two-way, Three-way, etc. ANOVA)

Fonctions et packages nécessaires : `leveneTest {car}` ; `TukeyHSD {stats}` ; `etaSquared {lsr}` ; `interaction.plot {stats}`

L'ANOVA à plusieurs facteurs intersujets permet de tester l'existence d'une différence significative entre les moyennes d'une variable quantitative (VD) provenant de plusieurs groupes formés par plusieurs variables nominales (ou facteurs, VI). La différence par rapport au test présenté au chapitre précédent est que vous évaluez l'effet de plusieurs facteurs à la fois sur la même variable quantitative et non plus l'effet d'un seul. Le test nous donne ainsi les effets principaux de chaque facteur (comme dans l'ANOVA à un facteur), mais en plus aux effets d'interaction entre ces facteurs.

Exemple : Nous souhaitons savoir s'il existe un effet conjoint du *type d'entretien psychologique* (VI 1, `EPsy`) et du *sexe* (VI 2, `Sexe`) sur le *nombre problèmes médicaux* (VD, `pb_med`) chez les adolescent·es, variables tirées de la base de données `Database`. Nous avons une variable nominale du type d'entretien à trois modalités et la variable nominale du sexe à deux modalités : l'ANOVA est donc de type 3X2.

**Attention** : Pour que les commandes suivantes soient exécutées correctement, il est nécessaire que les VI soient bel et bien considérées comme des facteurs par R (cf. chapitre 4.2.7). De plus, afin de faciliter la lecture des tableaux, renommez les modalités de vos facteurs avec le label approprié (cf. chapitre 4.2.8).

Dans les conditions d'application du test, l'**homogénéité des variances** de nos différents groupes doit être vérifiée grâce au **test de Levene**.

```
leveneTest(pb_med ~ EPsy * Sexe, data=Database, center=mean)
```

```
Levene's Test for Homogeneity of Variance (center = mean)
Df F value Pr(>F)
group 5 0.8471 0.5199
93
```

Après cela, vous devez créer une nouvelle variable qui est l'analyse de l'ANOVA pour finalement en demander un résumé : cela vous permettra d'avoir les informations nécessaires pour savoir s'il existe des effets principaux et/ou d'interaction.

```
ANOVA_PbyPS <- aov(pb_med ~ EPsy * Sexe, data=Database)
summary(ANOVA_PbyPS)
```

```
            Df Sum Sq Mean Sq F value    Pr(>F)
EPsy        2   56.4   28.198   6.452 0.00238 ***
Sexe        1     0.9    0.853   0.195 0.65961
EPsy:Sexe   2     8.4    4.190   0.959 0.38718
Residuals  93  406.5    4.370
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

**Remarque** : Nous savons *seulement* qu'il existe un effet principal du type d'entretien psychologique, mais ne savons pas combien de différences existent, ni ce qu'elles sont. Pour savoir où se situent ces différences, vous devez utiliser **un test post-hoc**, comme le test **HSD de Tukey**. Dans le cas où l'ANOVA n'indique aucun effet significatif, ou les si les facteurs n'ont que deux modalités, il n'est pas nécessaire de poursuivre avec un test de Tukey HSD.

```
TukeyHSD(ANOVA_PbyPS)
```

```

Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = pb_med ~ EPsy * Sexe, data = Database)

$EPsy
    diff      lwr      upr     p adj
G-L 1.5708333  0.4119584 2.729708 0.0048519
P-L 1.7761905  0.3594662 3.192915 0.0100101
P-G 0.2053571 -1.0974040 1.508118 0.9253080

$Sexe
    diff      lwr      upr     p adj
Femme-Homme 0.1857757 -0.6815751 1.053127 0.6715759

$`EPsy:Sexe`
    diff      lwr      upr     p adj
G:Homme-L:Homme 1.5738462 -0.1302087 3.277901 0.0874600
P:Homme-L:Homme 1.1700000 -0.9664631 3.306463 0.6048082
L:Femme-L:Homme -0.2800000 -3.2603055 2.700305 0.9997870
G:Femme-L:Homme 1.4654545 -0.3129180 3.243827 0.1676746
P:Femme-L:Homme 2.4755556  0.1107069 4.840404 0.0346635
P:Homme-G:Homme -0.4038462 -2.5269423 1.719250 0.9936571
L:Femme-G:Homme -1.8538462 -4.8245840 1.116892 0.4607179
G:Femme-G:Homme -0.1083916 -1.8706832 1.653900 0.9999737
P:Femme-G:Homme 0.9017094 -1.4510702 3.254489 0.8739981
L:Femme-P:Homme -1.4500000 -4.6882024 1.788202 0.7826669
G:Femme-P:Homme 0.2954545 -1.8877411 2.478650 0.9987389
P:Femme-P:Homme 1.3055556 -1.3770260 3.988137 0.7171516
G:Femme-L:Femme 1.7454545 -1.2685278 4.759437 0.5449354
P:Femme-L:Femme 2.7555556 -0.6376715 6.148783 0.1801874
P:Femme-G:Femme 1.0101010 -1.3970504 3.417252 0.8254951

```

Grâce au test post-hoc, vous pouvez désormais connaître les différences significatives de la VD en fonction des modalités des facteurs. Ces comparaisons *a posteriori* sont également effectuées au sein de l'interaction (ici, entre le type d'entretien psychologique et le sexe). Les effets d'interaction sont notés de la manière « Fact1:Fact2 » (en l'occurrence « EPsy:Sexe »). Les comparaisons au sein de l'effet d'interaction peuvent être relativement nombreuses suivant le nombre de modalités des facteurs.

Un effet significatif peut être détecté grâce à des échantillons très importants, mais sans grands écarts dans les moyennes. La **taille d'effet** renseigne sur la grandeur des effets significatifs observés. Les tailles d'effets communément utilisées dans l'analyse de variance sont le  $\eta^2$  (éta-carré) et le  $\eta^2$  partiel.

#### **etaSquared(ANOVA\_PbyPS)**

	eta.sq	eta.sq.part
EPsy	0.103616671	0.107419863
Sexe	0.001807612	0.002095087
EPsy:Sexe	0.017749518	0.020199105

Comme nous avons plusieurs VIs, les tailles d'effet individuelles ne permettent pas de connaître le pourcentage de la variance de la VD expliquée par l'ensemble de nos VIs. Pour le savoir, il peut être utile de calculer le **R<sup>2</sup>** (variance expliquée dans l'échantillon) ou **R<sup>2</sup> ajusté** (pourcentage de variance ajusté à la population). Il est à votre convenance de choisir l'un des deux indices pour autant que vous le précisiez. Vous remarquerez que la fonction **lm(...)** est celle utilisée pour la régression linéaire (cf. chapitre 4.4.8).

#### **summary(lm(ANOVA\_PbyPS))**

```

Call:
lm(formula = ANOVA_PbyPS)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.6538 -1.6047 -0.5455  1.4444  5.9200 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.0800    0.4181   4.975 2.98e-06 ***
EPsyG        1.5738    0.5856   2.688  0.00853 **  
EPsyP        1.1700    0.7342   1.594  0.11442    
SexeFemme   -0.2800    1.0242  -0.273  0.78516    
EPsyG:SexeFemme  0.1716    1.1898   0.144  0.88563    
EPsyP:SexeFemme  1.5856    1.3779   1.151  0.25282    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.091 on 93 degrees of freedom
Multiple R-squared:  0.139,   Adjusted R-squared:  0.09273 
F-statistic: 3.003 on 5 and 93 DF,  p-value: 0.01474

```

Les résultats (p. ex.:  $F$ ,  $p$ ) de la fonction `lm(...)` sont identiques à ceux de l'ANOVA précédemment calculée. En effet, une partie des calculs de la régression linéaire sont effectués par l'intermédiaire d'une ANOVA (cf. chapitre 4.4.8).

**Rappel :** Pour obtenir les moyennes et écarts-types en fonction des modalités, référez-vous au chapitre *Statistiques descriptives* (4.3.10). Pour un graphique représentant ces moyennes et écarts-types, référez-vous au chapitre *Graphique d'interaction* (4.3.13.8).

```

# A tibble: 6 x 5
# Groups:   EPsy [3]
  EPsy Sexe Nombre Moyenne EcartType
  <fct> <fct> <int>   <dbl>     <dbl>
1 L     Homme    25     2.08     2.18
2 L     Femme     5      1.8      1.30
3 G     Homme    26     3.65     2.23
4 G     Femme    22     3.55     2.04
5 P     Homme    12     3.25     2.09
6 P     Femme     9      4.56     1.81

```

**Notation :** « Une ANOVA à facteurs intersujets 3 (Type d'entretien psychologique : L, G ou P) x 2 (Sexe : garçon ou fille) a été effectué afin d'évaluer l'importance des problèmes médicaux rencontrés par des adolescent·es. Il existe un effet principal du type d'entretien psychologique sur le nombre de problèmes médicaux,  $F(2, 93) = 6.45, p < .05, \eta_p^2 = .11$ . Les types d'entretien psychologique « P » ( $M = 3.81, SD = 2.04$ ) et « G » ( $M = 3.60, SD = 2.12$ ) ne diffèrent pas, mais sont significativement différents du type d'entretien psychologique « L » ( $M = 2.03, SD = 2.04$ ). Il n'existe aucun effet principal significatif du sexe des participant·es,  $F(1, 93) = .20, p > .05, \eta_p^2 < .01$ . Les participantes ( $M = 3.56, SD = 2.03$ ) et les participants ( $M = 2.95, SD = 2.27$ ) n'ont pas un nombre de problèmes médicaux significativement différents. L'effet d'interaction du sexe et du type d'entretien psychologique n'est pas significatif,  $F(2, 93) = .96, p > .05, \eta_p^2 = .02$ . L'ensemble des variables et l'effet d'interaction expliquent 14% de la variance du nombre de problèmes médicaux dans l'échantillon,  $R^2 = .14$ . »

#### 4.4.4.3 Préparation de la base de données pour l'ANOVA à mesures répétées

Ce chapitre présente comment préciser au logiciel R que deux ou plusieurs variables dépendantes constituent en réalité les modalités d'une ou plusieurs VI (facteurs) à mesures répétées dans notre base de données. Cette manipulation est essentielle avant de faire l'**ANOVA à mesures répétées et l'ANOVA mixte**.

Par exemple, une base de données peut contenir les colonnes « Score\_avant », « Score\_pendant » et « Score\_apres », représentant respectivement des scores avant, pendant et après un entraînement. De ce fait, les participant·es ont un score pour ces trois variables quantitatives, qui sont donc des mesures répétées. Initialement, le logiciel R ne suppose pas que ces variables représentent en réalité les trois modalités d'une variable indépendante à mesures répétées (variable pouvant être appelée « Période de test »). Il vous faut donc le lui indiquer.

##### 4.4.4.3.1 Préparation pour 1 variable à mesures répétées : gather {tidyR}

Exemple : Une base de données contient cinq variables quantitatives appelées `SuccesL` (jugement de succès pour des visages porteurs de lunettes), `IntellL` (jugement d'intelligence), `ConflL` (jugement de confiance), `AttraitL` (jugement d'attrait) et `HonnetL` (jugement d'honnêteté). Nous souhaitons transformer la base de données de telle sorte que ces cinq modalités soient comprises dans une seul variable indépendante appelée `Traits` mesurée par un score `ScoreJugement`, variables tirées de la base de données `Database`.

Toutes les étapes sont nécessaires. La première sera de **sélectionner les variables d'intérêt** (non seulement les cinq VDs, mais également les groupes, si vous souhaitez effectuer une ANOVA mixte, avec à la fois des groupes indépendants *et* appariés) **ainsi qu'une variable d'identification** (l'ID, le numéro de participant·e, etc.) dans une nouvelle base de données.

```
DatabaseT <- Database[ ,c("Numparticipant", "EthnieVisages",
  "SuccesL", "IntellL", "ConflL", "AttraitL", "HonnetL")]
```

Dans le cas présent, nous allons créer une base de données avec « Numparticipant » (en guise de variable d'identification), « EthnieVisages » (groupes indépendants à deux modalités) et les cinq variables quantitatives.

**Remarque** : Il est tout à fait possible d'inscrire le numéro des colonnes (information obtenue en maintenant la souris sur la colonne désirée dans l'aperçu de la base de données) à la place du nom des colonnes. En voici l'exemple :

```
DatabaseT <- Database[ ,c(1,2,61,63,65,67,69)]
```

La fonction permettant d'exécuter l'ANOVA à mesures répétées et mixte ne tolère pas les valeurs manquantes. Ces valeurs manquantes peuvent être présentes initialement dans la base de données, mais également après avoir supprimé des valeurs extrêmes et aberrantes. Afin d'éviter toute erreur durant la compilation, il est nécessaire de **supprimer les lignes comportant des valeurs manquantes** de la base de données `DatabaseT` avant toute modification supplémentaire.

```
DatabaseT <- DatabaseT[complete.cases(DatabaseT),]
```

La fonction ci-dessous permet la création de la base de données finale. La fonction comprend plusieurs éléments dont voici l'ordre :

- **La base de données à transformer**, en l'occurrence `DatabaseT`.
- **Le nom (à choix) de la variable indépendante** qui comprendra les différentes modalités, en l'occurrence `Traits`.
- **Le nom (à choix) de la variable quantitative** qui correspondra aux différentes valeurs qu'ont ces modalités, en l'occurrence `ScoreJugement`.
- **Les modalités à considérer dans cette nouvelle variable indépendante**, en y inscrivant leurs noms, en l'occurrence `c(SuccesL, IntellL, ...)`.

```
DatabaseT_lg <- gather(DatabaseT, Traits, ScoreJugement,
                        c(SuccesL, IntellL, ConfL, AttraitL, HonnetL),
                        factor_key=TRUE)
View(DatabaseT_lg) #Controle
```

	Numparticipant Numéro du participant	EthnieVisages EthnieVisagesBlancNoir	Traits	ScoreJugement
98	98	2	SuccesL	5.250000
99	99	1	SuccesL	4.250000
100	100	1	SuccesL	4.500000
101	101	2	SuccesL	3.750000
102	102	2	SuccesL	5.500000
103	103	2	SuccesL	5.750000
104	1	2	IntellL	6.000000
105	2	2	IntellL	5.500000
106	3	2	IntellL	6.500000
107	4	2	IntellL	5.750000
108	5	1	IntellL	3.750000

Ci-dessus, nous remarquons que la base de données finale a désormais le nom de la variable indépendante regroupant les cinq modalités sous « Traits », ainsi que leurs valeurs respectives sous « ScoreJugement ». L’importance de la variable d’identification (en l’occurrence « Numparticipant ») réside dans le fait que les **participant·es se retrouvent sur plusieurs lignes différentes**, contrairement à l’accoutumée. Ceci permet alors de savoir quelle valeur un·e participant·e a sur chacune de ces modalités en l’identifiant grâce à son numéro.

Notez que la base de données finale est  $x$  fois plus longue que l’originale (où  $x$  est le nombre de modalités). Dans cet exemple, la base de données originale contenait 103 lignes, et la base de données finale en contient 515 (soit  $103 \times 5$ ).

#### 4.4.4.3.2 Préparation pour 2 variables (ou plus) à mesures répétées

Fonctions et packages nécessaires : *gather*, *separate* {tidyR}

Exemple : Une base de données contient dix variables quantitatives appelées *SuccesL* (jugement de succès pour visages porteurs de lunettes), *IntellL* (jugement d'intelligence), *ConfL* (jugement de confiance), *AttraitL* (jugement d'attrait), *HonnetL* (jugement d'honnêteté), *SuccessSL* (jugement pour visages non porteurs de lunettes), *IntellSL*, *ConfSL*, *AttraitSL* et *HonnetSL*, variables tirées de la base de données *Database*.

Contrairement au chapitre précédent, deux variables indépendantes sont imbriquées, soit les traits de jugements (première partie du nom, *Succes*) et le port de lunettes (seconde partie du nom, *L/SL*). Ainsi, nous souhaitons une première variable indépendante à 5 modalités (appelée *Traits*) et une seconde variable indépendante à 2 modalités (appelée *Lunettes*). Pour plus d'informations sur cette étape, veuillez vous référer au chapitre précédent.

```
DatabaseTL <- Database[, c(1, 2, 60:69)]  
DatabaseTL <- DatabaseTL[complete.cases(DatabaseTL), ]
```

La création de la base de données finale comporte **deux étapes** : une première regroupant les 10 variables sous une seule colonne, puis la seconde permettant de les distinguer sous deux colonnes respectives.

**Attention** : Si le nom de vos variables ne comporte aucun symbole (comme « \_ », « - », « . », ou autres) séparant les deux modalités, alors il est nécessaire de les renommer (voir chapitre *Renommer une variable* (4.2.3)). Si ce n'est pas le cas, la prochaine commande n'est pas à exécuter.

Dans notre exemple, il est nécessaire de séparer les deux modalités par un symbole, prenons « \_ », pour transformer, entre autres, « SuccessSL » en « Succes\_SL », afin de séparer les deux facteurs. En voici deux exemples (parmi les dix à effectuer) :

```
names(DatabaseTL)[names(DatabaseTL) == "SuccessSL"] <- "Succes_SL"  
names(DatabaseTL)[names(DatabaseTL) == "IntellSL"] <- "Intell_SL"
```

**La première étape** consiste à rassembler toutes les variables sous une seule colonne *key*.

La fonction ci-dessous permet la création de la **base de données temporaire**. La fonction comprend plusieurs éléments, sous *gather*, dont voici l'ordre :

- **La variable de rassemblement** en l'occurrence *key*.
- **Le nom (à choix) de la variable dépendante** qui correspondra aux différentes valeurs qu'ont ces modalités imbriquées, en l'occurrence *ScoreJugement*.
- **Les variables à ne pas considérer dans la fusion, précédées du symbole « - »** qui correspondent à la variable d'identification, ici *-NumParticipant*, et aux autres variables, ici *-EthnieVisages*.

```
DatabaseTL_lg <- DatabaseTL %>%  
  gather(key, ScoreJugement, -NumParticipant, -EthnieVisages)  
DatabaseTL_lg %>% head(8) #Contrôle
```

**La seconde étape** consiste à attribuer les modalités à leurs variables indépendantes respectives.

La fonction ci-dessous permet la création de la **base de données finale**. La fonction comprend plusieurs éléments, sous `separate`, dont voici l'ordre :

- **Les deux noms (à choix) de vos variables indépendantes** dont l'ordre dépend du nom des variables quantitatives, en l'occurrence `Traits` et `Lunettes`.
- **Le symbole séparant les deux modalités** précédé de deux backslashes, en l'occurrence `"\_"`.

```
DatabaseTL_lg <- DatabaseTL_lg %>%  
  separate(key, into = c("Traits", "Lunettes"), sep = "\\_")  
DatabaseTL_lg %>% head(8) #Contrôle
```

	Numparticipant Numéro du participant	EthnieVisages EthnieVisagesBlancNoir	Traits	Lunettes	ScoreJugement
	617	102	2 Conf	L	4.250000
	618	103	2 Conf	L	4.000000
	619	1	2 Attrait	SL	5.250000
	620	2	2 Attrait	SL	4.250000
	621	3	2 Attrait	SL	3.500000

#### 4.4.4.4 ANOVA à mesures répétées (Repeated Measures ANOVA)

Fonctions et packages nécessaires : `ezANOVA`, `ezStats`, `ezPlot {ez}` ; `aov {stats}` ; `PostHocTest {DescTools}`

L'ANOVA à mesures répétées permet de tester l'existence d'effets significatifs d'une ou plusieurs VI nominale(s) à mesures répétées (= un ou plusieurs facteur(s) intrasujets(s)).

Exemple : Nous souhaitons savoir s'il existe des effets du *trait* (`Traits`, intrasujet) et du *port de lunettes* (`Lunettes`, intrasujet) sur le *jugement porté à des visages* (VD, `ScoreJugement`), variables tirées de la base de données `Database_lg`. Nous avons une variable nominale du trait à cinq modalités (*attrait, honnêteté, intelligence, succès professionnel et confiance*) et une variable nominale du port de lunettes à deux modalités (*avec et sans*). La variable dépendante quantitative est le score de jugement porté à ces visages. Précisons que les participant·es ont à la fois jugé des visages *avec lunettes* et *sans lunettes* et sur les *cinq différents traits* (d'où la notion de *mesures répétées 5x2* ou, en anglais, *5x2 repeated measures ANOVA*).

Attention : Avant de continuer, il est obligatoire de préparer la base de données pour l'ANOVA à mesures répétées. Toutes les étapes sont décrites dans le chapitre précédent *Préparation de la base de données pour l'ANOVA à mesures répétées* (4.4.4.3).

La fonction `ezANOVA` permet de lancer l'analyse de manière simplifiée, mais également de considérer la condition d'application de la **sphéricité**, mesurée par le test de **Mauchly**, et les corrections en cas de non-respect de la sphéricité (correction de Greenhouse-Geisser). La fonction comprend plusieurs paramètres dont voici l'ordre.

- **La base de données** sous `data=`, en l'occurrence `Database_lg`.
- **La variable dépendante** sous `dv=`, en l'occurrence `.(ScoreJugement)`.
- **La variable d'identification** sous `wid=`, en l'occurrence `.(Numparticipant)`.
- **La (ou les) facteur(s) intrasujets** sous `within=`, en l'occurrence `.(Traits, Lunettes)`.

```
ANOVAmr_TL <- ezANOVA(data=Database_lg,
                         dv = .(ScoreJugement),
                         wid = .(Numparticipant),
                         within = .(Traits, Lunettes),
                         detailed = TRUE, type = 3)
```

**ANOVAmr\_TL**

	Effect	DFn	DFd	SSn	SSd	F	p	p<.05	ges
1	(Intercept)	1	102	17885.41684	393.13941	4640.37044	7.312145e-87	*	0.9564552
2	Traits	4	408	251.31129	215.61510	118.88662	3.876197e-67	*	0.2358433
3	Lunettes	1	102	24.88027	95.77598	26.49711	1.287190e-06	*	0.0296492
4	Traits:Lunettes	4	408	68.41582	109.74391	63.58816	9.374454e-42	*	0.0775083

\$`ANOVA`				Effets principaux et effets d'interaction			
Effect	DFn	DFd					
1 (Intercept)	1	102					
2 Traits	4	408					
3 Lunettes	1	102					
4 Traits:Lunettes	4	408					

\$`Mauchly's Test for Sphericity`				Test de Mauchly (sphéricité)			
Effect	W		p	p<.05			
2 Traits	0.3465651	8.082936e-19	*				
4 Traits:Lunettes	0.5235695	1.469800e-10	*				

\$`Sphericity Corrections`				Correction de sphéricité (Greenhouse-Geisser)			
Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]	p[HF]<.05	
2 Traits	0.7053193	3.518761e-48	*	0.7274533	1.323203e-49	*	
4 Traits:Lunettes	0.7514172	4.888600e-32	*	0.7767714	4.981885e-33	*	

Le tableau ci-dessus donne les résultats de l'analyse, les effets principaux et d'interaction, le test de Mauchly (sphéricité) et la correction de la sphéricité en cas de non-respect de la condition d'application. En vert, les degrés de liberté (respectivement, du numérateur et du dénominateur) de l'indice  $F$ . En rouge, la valeur de l'indice  $F$ . En orange, la valeur  $p$ . Finalement, en bleu, les tailles d'effet générales par l'indice  $\eta_g^2$  (eta carré général, adapté aux mesures répétées).

Concernant le **test de sphéricité de Mauchly**, celui-ci **doit être non significatif** pour considérer l'égalité des covariances des erreurs du ou des facteur(s) intrasujet(s). Ce test n'est exécuté que pour les facteurs intrasujets comportant plus de deux modalités (ici l'effet principal Traits et l'interaction Traits:Lunettes). Dans notre exemple, les sphéricités ne sont pas respectées. Pour cette raison, nous devons nous référer à la correction de Greenhouse-Geisser qui nous indique si ces deux effets sont significatifs (mais nous notons dans la phrase de report de l'analyse le F habituel en indiquant que la correction de Greenhouse-Geisser a été effectuée).

**Les comparaisons *a posteriori*** ou **tests post-hoc** permettent de savoir où se situent les différences significatives dans nos effets de variables à plus de deux modalités. Il est nécessaire de recréer une variable d'ANOVA (sans inclure l'erreur des effets intrasujets) pour ensuite lancer les analyses post-hoc. La fonction `PostHocTest` comprend plusieurs paramètres dont voici l'ordre :

- **Les facteurs à traiter** sous `which=`, en l'occurrence `which="Traits"`. Ceci signifie que nous ne souhaitons que les tests post-hoc de la variable « Traits », et non de l'effet d'interaction. Si vous souhaitez la totalité des variables, inscrivez `which=NULL`.
- **Le test post-hoc désiré** sous `method=`, en l'occurrence `method="hsd"`, représentant le test de Tukey HSD. Il est, entre autres, possible d'utiliser le test de Bonferroni sous "bonferroni" et le test de LSD sous "lsd".

```
ANOVAmr_TLpost <- aov(ScoreJugement ~ Traits * Lunettes,
                        data=Database_lg)
PostHocTest(ANOVAmr_TLpost, which="Traits",
            method="hsd", conf.level=.95)
```

```
Posthoc multiple comparisons of means : Tukey HSD
 95% family-wise confidence level

$`Traits`
    diff      lwr.ci     upr.ci   pval
Conf-Attrait  1.0000000  0.75942377 1.2405762 < 2e-16 ***
Honnet-Attrait  0.9012945  0.66071827 1.1418707 < 2e-16 ***
Intell-Attrait  1.4097896  1.16921342 1.6503659 < 2e-16 ***
Succes-Attrait  1.2762945  1.03571827 1.5168707 < 2e-16 ***
Honnet-Conf   -0.0987055 -0.33928173 0.1418707 0.79538
Intell-Conf    0.4097896  0.16921342 0.6503659 3.6e-05 ***
Succes-Conf    0.2762945  0.03571827 0.5168707 0.01503 *
Intell-Honnet  0.5084951  0.26791892 0.7490714 1.0e-07 ***
Succes-Honnet  0.3750000  0.13442377 0.6155762 0.00022 ***
Succes-Intell -0.1334951 -0.37407137 0.1070811 0.55202

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Comme nous avons plusieurs VIs, les tailles d'effet individuelles ne permettent pas de connaître le pourcentage de la variance de la VD expliquée par l'*ensemble* de nos VIs. Il peut être utile pour le savoir de calculer les  **$R^2$**  (variance expliquée dans l'échantillon) ou  **$R^2$  ajusté** (pourcentage de variance ajusté à la population).

```
summary(lm(ANOVAmr_TLpost))
```

```
Residual standard error: 0.8935 on 1020 degrees of freedom
Multiple R-squared:  0.2974, Adjusted R-squared:  0.2912
F-statistic: 47.96 on 9 and 1020 DF,  p-value: < 2.2e-16
```

Pour obtenir **les statistiques descriptives** en fonction de modalités désirées, la fonction `ezStats` se trouve plus simple d'utilisation que la fonction `group_by` lorsque vous effectuez une ANOVA à mesures répétées ou mixte, car elle ressemble aux commandes de `ezANOVA`. Pour ce faire, précisez les facteurs désirés. Pour chaque effet principal, n'entrez que ce facteur dans la commande `within =`, et mettez plusieurs facteurs pour les effets d'interaction (comme ci-dessous `Traits`, `Lunettes` ; pour obtenir les trois tableaux ci-dessous, nous lancerons 3 fois la commande `ezStats`). Il est toujours nécessaire d'inscrire la variable d'identification.

```
ezStats(data = Database_lg,
        dv = .(ScoreJugement),
        wid = .(Numparticipant),
        within = .(Traits, Lunettes))
```

	Traits	Lunettes	N	Mean	SD	FLSD
1	Attrait	L	103	2.979773	1.0354860	0.1420675
2	Attrait	SL	103	3.519417	1.0382922	0.1420675
3	Conf	L	103	4.319579	0.8222380	0.1420675
4	Conf	SL	103	4.179612	0.8242774	0.1420675
5	Honnet	L	103	4.282362	0.8850945	0.1420675
6	Honnet	SL	103	4.019417	0.8819328	0.1420675
7	Intell	L	103	5.066343	0.9419681	0.1420675
8	Intell	SL	103	4.252427	0.7991780	0.1420675
9	Succes	L	103	4.964401	0.8526840	0.1420675
10	Succes	SL	103	4.087379	0.8230062	0.1420675

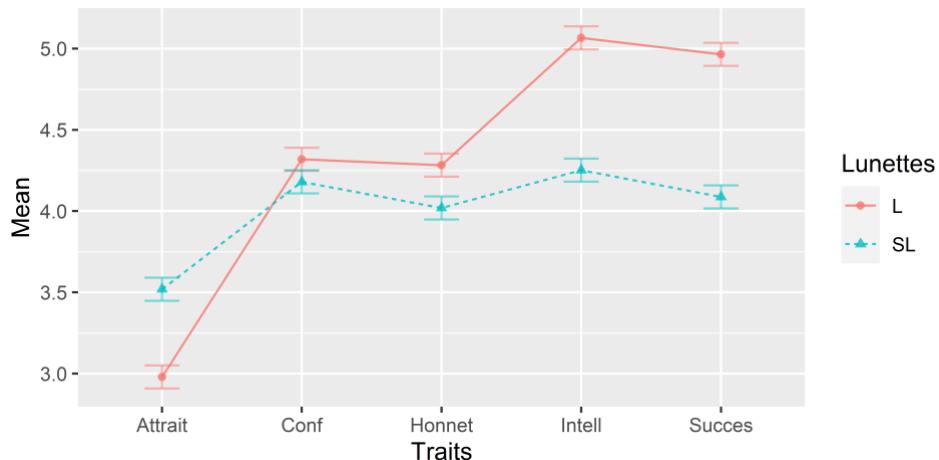
	Traits	N	Mean	SD	FLSD
1	Attrait	103	3.249595	0.9474462	0.1408086
2	Conf	103	4.249595	0.7030148	0.1408086
3	Honnet	103	4.150890	0.7562178	0.1408086
4	Intell	103	4.659385	0.7237233	0.1408086
5	Succes	103	4.525890	0.7046742	0.1408086

	Lunettes	N	Mean	SD	FLSD
1	L	103	4.322492	0.7074329	0.1197763
2	SL	103	4.011650	0.6769020	0.1197763

Pour savoir comment se présente **graphiquement l'effet d'interaction**, la fonction `ezPlot` est simple d'utilisation. Suivant la même logique que les fonctions `ezANOVA` et `ezStats`, vous pouvez préciser l'interaction désirée. La fonction `ezPlot` comprend plusieurs paramètres supplémentaires dont voici l'ordre :

- **Le facteur présenté en abscisse (axe x)** sous `x=`, en l'occurrence `x = .(Traits)`. Ce facteur peut être interverti avec les autres paramètres.
- **Le facteur présenté en légendes (séparation en couleurs)** sous `split`, en l'occurrence `split = .(Lunettes)`. Ce facteur peut être interverti avec les autres paramètres.
- **Le(s) facteur(s) scindant le graphique en lignes et/ou en colonnes (si besoin)** sous `row` et `col`, respectivement. Ce paramètre est nécessaire si vous souhaitez explorer graphiquement un effet d'interaction de trois à quatre facteurs. Pour une illustration de ce paramètre, référez-vous au chapitre suivant.

```
ezPlot(data = Database_lg,
       dv = .(ScoreJugement),
       wid = .(Numparticipant),
       within = .(Traits, Lunettes),
       x = .(Traits), split = .(Lunettes),
       do_lines = TRUE)
```



**Notation :** « Une ANOVA à mesures répétées 5 (Traits : succès, intelligence, confiance, attrait et honnêteté) x 2 (Port de lunettes : avec et sans) a été effectué afin d'évaluer l'effet de ces variables sur le jugement des visages.

Il existe un effet principal significatif du trait de jugement sur l'évaluation de visages,  $F(4, 408) = 118.89, p < .05, \eta_g^2 = .24$  (avec correction Greenhouse-Geisser). Selon le test post-hoc de Tukey HSD ( $p < .05$ ), l'évaluation de l'intelligence ( $M = 4.66, SD = .94$ ) ne diffère pas de l'évaluation du succès ( $M = 4.53, SD = .94$ ), ces dernières étant supérieures à l'évaluation de la confiance ( $M = 4.25, SD = .82$ ) égale à l'évaluation de l'honnêteté ( $M = 4.15, SD = .89$ ), ces dernières étant supérieures à l'évaluation de l'attrait ( $M = 3.24, SD = 1.07$ ).

Il existe un effet principal significatif du port de lunettes sur l'évaluation de visages,  $F(1, 102) = 26.50, p < .05, \eta_g^2 = .03$ . Les visages portant des lunettes ( $M = 4.32, SD = 1.17$ ) obtiennent une évaluation des traits globalement plus élevée que les visages ne portant pas de lunettes ( $M = 4.01, SD = .91$ ).

Un effet d'interaction significatif est observé entre les traits et le port de lunettes sur le jugement de visages,  $F(4, 408) = 63.59, p < .05, \eta_g^2 = .08$  (avec correction Greenhouse-Geisser). L'effet principal du port de lunettes est inversé pour l'évaluation de l'attrait, inexistant pour celle de la confiance et de l'honnêteté, mais valable pour celle de l'intelligence et du succès. L'ensemble des variables et l'effet d'interaction expliquent 29% de la variance du jugement de visages dans la population,  $R^2 = .29$ . »

#### 4.4.4.5 ANOVA mixte (mesures répétées et groupes indépendants)

Fonctions et packages nécessaires : `leveneTest {car}` ; `boxM {heplots}` ; `ezANOVA, ezStats, ezPlot {ez}` ; `aov {stats}` ; `PostHocTest {DescTools}`

L'ANOVA mixte permet de tester l'existence l'effet d'une ou plusieurs VI nominale(s) à mesures répétées (= un ou plusieurs facteur(s) intrasujets(s)) ainsi que de VI nominale(s) à groupes indépendants (= un ou plusieurs facteurs(s) intersujet(s)) sur une VD.

Exemple : Nous souhaitons savoir s'il existe des effets du *trait* (`Traits`, intrasujet), du *port de lunettes* (`Lunettes`, intrasujet) et de l'*ethnie des visages présentés* (`EthnieVisages`, intersujet) sur le *jugement porté à des visages* (VD, `ScoreJugement`), variables tirées de la base de données `Database_1g`. Nous reprenons à l'identique l'exemple du chapitre précédent, mais en y ajoutant une variable nominale de l'ethnie des visages à deux modalités (*visages noirs* ou *visages blancs*). Nous ne sommes plus dans un cas d'ANOVA à mesures répétées, mais bien de celui d'une ANOVA mixte.

Attention : Avant de continuer, il est obligatoire de préparer la base de données pour l'ANOVA à mesures répétées. Toutes les étapes sont décrites dans le chapitre précédent *Préparation de la base de données pour l'ANOVA à mesures répétées* (4.4.4.3).

Selon les conditions d'application du test, l'**homogénéité des variances** de nos différents groupes doit être vérifiée grâce au **test de Levene**, devant être exécuté pour chaque variable quantitative individuellement, cela en utilisant la base de données originale `Database`.

```
leveneTest(IntellSL~EthnieVisages, data=Database, center=mean)
leveneTest(IntellL ~EthnieVisages, data=Database, center=mean)
# Exemples pour IntellSL et IntellL
```

Nous devons également vérifier l'**homogénéité de la matrice de variances/covariances grâce au test de Box**, qui doit être non significatif pour considérer cette condition d'application comme respectée. Toutes les variables quantitatives doivent être insérées ainsi que le(s) facteur(s).

```
boxM(cbind(SuccesSL, SuccesL, IntellSL, IntellL, ConfSL, ConfL,
            AttraitSL, AttraitL, HonnetSL, HonnetL) ~ Sexe,
      data=Database)
```

```
Box's M-test for Homogeneity of Covariance Matrices
data: Y
Chi-Sq (approx.) = 66.408, df = 55, p-value = 0.1394
```

Dans le cas présent, l'homogénéité de la matrice de variances/covariances peut être considérée respectée,  $\chi^2(55) = 66.41, p > .05$ .

Pour vérifier la condition d'application du **nombre égal de participant·es entre les groupes**, vous pouvez simplement utiliser la fonction `table`. Lorsque vos groupes sont inégaux (même avec 1 seule observation de différence), vous obtiendrez un message d'avertissement en lançant la commande `ezANOVA`. Cependant, ces commandes resteront toujours fonctionnelles et le message n'impactera pas vos analyses.

```
table(Database$EthnieVisages)
```

```
EthnieVisages
 1 2
49 54
```

La fonction `ezANOVA` permet de lancer l'analyse, et également de considérer la condition d'application de la **sphéricité**, mesurée par le test de **Mauchly**, et les corrections en cas de non-respect de la sphéricité (correction de Greenhouse-Geisser). La fonction comprend plusieurs éléments, dont voici l'ordre.

- **La base de données** sous `data=`, en l'occurrence `Database_lg`.
- **La variable dépendante** sous `dv=`, en l'occurrence `.(ScoreJugement)`.
- **La variable d'identification** sous `wid=`, en l'occurrence `.(Numparticipant)`.
- **La (ou les) facteur(s) intrasujets** sous `within=`, en l'occurrence `.(Traits, Lunettes)`.
- **Le(s) facteur(s) intersujets** sous `between=`, en l'occurrence `(EthnieVisages)`.

```
ANOVAmixt_TLE <- ezANOVA(data = Database_lg,
                           dv = .(ScoreJugement),
                           wid = .(Numparticipant),
                           within = .(Traits, Lunettes),
                           between = .(EthnieVisages),
                           detailed = TRUE, type = 3)
```

`ANOVAmixt_TLE`

A la création de l'ANOVA un message d'avertissement sera donné, précisant si tel est le cas que l'égalité en nombre de participant·es n'est pas respectée entre les différents groupes du facteur intersujet. Nous pouvons constater, malgré l'avertissement, que l'égalité n'est à peu près respectée entre les groupes (tant que  $2 \cdot n_1 > n_2$ , on peut dire que l'inégalité n'est pas trop forte).

Warning: Data is unbalanced (unequal N per group). Make sure you specified a well-considered value for the type argument to ezANOVA().																																																																																																																							
> ANOVAmixt_TLE																																																																																																																							
\$`ANOVA`																																																																																																																							
<table border="1"> <thead> <tr> <th></th> <th></th> <th>Effect</th> <th>DFn</th> <th>DFd</th> <th>SSn</th> <th>SSd</th> <th>F</th> <th>p</th> <th>p&lt;.05</th> </tr> </thead> <tbody> <tr> <td>1</td> <td></td> <td>(Intercept)</td> <td>1</td> <td>101</td> <td>1.782902e+04</td> <td>391.93158</td> <td>4.594503e+03</td> <td>5.030527e-86</td> <td>*</td> </tr> <tr> <td>2</td> <td></td> <td>EthnieVisages</td> <td>1</td> <td>101</td> <td>1.207834e+00</td> <td>391.93158</td> <td>3.112563e-01</td> <td>5.781448e-01</td> <td>9.571759e-01</td> </tr> <tr> <td>3</td> <td></td> <td>Traits</td> <td>4</td> <td>404</td> <td>2.466353e+02</td> <td>206.59656</td> <td>1.205740e+02</td> <td>1.327839e-67</td> <td>1.511911e-03</td> </tr> <tr> <td>5</td> <td></td> <td>Lunettes</td> <td>1</td> <td>101</td> <td>2.477922e+01</td> <td>95.76829</td> <td>2.613288e+01</td> <td>1.515061e-06</td> <td>2.361715e-01</td> </tr> <tr> <td>4</td> <td></td> <td>EthnieVisages:Traits</td> <td>4</td> <td>404</td> <td>9.018546e+00</td> <td>206.59656</td> <td>4.408947e+00</td> <td>1.686774e-03</td> <td>3.012854e-02</td> </tr> <tr> <td>6</td> <td></td> <td>EthnieVisages:Lunettes</td> <td>1</td> <td>101</td> <td>7.696796e-03</td> <td>95.76829</td> <td>8.117263e-03</td> <td>9.283895e-01</td> <td>1.117970e-02</td> </tr> <tr> <td>7</td> <td></td> <td>Traits:Lunettes</td> <td>4</td> <td>404</td> <td>6.631181e+01</td> <td>103.37430</td> <td>6.478876e+01</td> <td>2.664109e-42</td> <td>9.648996e-06</td> </tr> <tr> <td>8</td> <td></td> <td>EthnieVisages:Traits:Lunettes</td> <td>4</td> <td>404</td> <td>6.369606e+00</td> <td>103.37430</td> <td>6.223309e+00</td> <td>7.229843e-05</td> <td>7.675133e-02</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>*</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>7.921998e-03</td> </tr> </tbody> </table>												Effect	DFn	DFd	SSn	SSd	F	p	p<.05	1		(Intercept)	1	101	1.782902e+04	391.93158	4.594503e+03	5.030527e-86	*	2		EthnieVisages	1	101	1.207834e+00	391.93158	3.112563e-01	5.781448e-01	9.571759e-01	3		Traits	4	404	2.466353e+02	206.59656	1.205740e+02	1.327839e-67	1.511911e-03	5		Lunettes	1	101	2.477922e+01	95.76829	2.613288e+01	1.515061e-06	2.361715e-01	4		EthnieVisages:Traits	4	404	9.018546e+00	206.59656	4.408947e+00	1.686774e-03	3.012854e-02	6		EthnieVisages:Lunettes	1	101	7.696796e-03	95.76829	8.117263e-03	9.283895e-01	1.117970e-02	7		Traits:Lunettes	4	404	6.631181e+01	103.37430	6.478876e+01	2.664109e-42	9.648996e-06	8		EthnieVisages:Traits:Lunettes	4	404	6.369606e+00	103.37430	6.223309e+00	7.229843e-05	7.675133e-02										*										7.921998e-03
		Effect	DFn	DFd	SSn	SSd	F	p	p<.05																																																																																																														
1		(Intercept)	1	101	1.782902e+04	391.93158	4.594503e+03	5.030527e-86	*																																																																																																														
2		EthnieVisages	1	101	1.207834e+00	391.93158	3.112563e-01	5.781448e-01	9.571759e-01																																																																																																														
3		Traits	4	404	2.466353e+02	206.59656	1.205740e+02	1.327839e-67	1.511911e-03																																																																																																														
5		Lunettes	1	101	2.477922e+01	95.76829	2.613288e+01	1.515061e-06	2.361715e-01																																																																																																														
4		EthnieVisages:Traits	4	404	9.018546e+00	206.59656	4.408947e+00	1.686774e-03	3.012854e-02																																																																																																														
6		EthnieVisages:Lunettes	1	101	7.696796e-03	95.76829	8.117263e-03	9.283895e-01	1.117970e-02																																																																																																														
7		Traits:Lunettes	4	404	6.631181e+01	103.37430	6.478876e+01	2.664109e-42	9.648996e-06																																																																																																														
8		EthnieVisages:Traits:Lunettes	4	404	6.369606e+00	103.37430	6.223309e+00	7.229843e-05	7.675133e-02																																																																																																														
									*																																																																																																														
									7.921998e-03																																																																																																														
Effets principaux et effets d'interaction																																																																																																																							
<table border="1"> <thead> <tr> <th></th> <th></th> <th>Effect</th> <th>W</th> <th>p</th> <th>p&lt;.05</th> </tr> </thead> <tbody> <tr> <td>3</td> <td></td> <td>Traits</td> <td>0.3564105</td> <td>4.864813e-18</td> <td>*</td> </tr> <tr> <td>4</td> <td></td> <td>EthnieVisages:Traits</td> <td>0.3564105</td> <td>4.864813e-18</td> <td>*</td> </tr> <tr> <td>7</td> <td></td> <td>Traits:Lunettes</td> <td>0.5834814</td> <td>2.302673e-08</td> <td>*</td> </tr> <tr> <td>8</td> <td></td> <td>EthnieVisages:Traits:Lunettes</td> <td>0.5834814</td> <td>2.302673e-08</td> <td>*</td> </tr> </tbody> </table>												Effect	W	p	p<.05	3		Traits	0.3564105	4.864813e-18	*	4		EthnieVisages:Traits	0.3564105	4.864813e-18	*	7		Traits:Lunettes	0.5834814	2.302673e-08	*	8		EthnieVisages:Traits:Lunettes	0.5834814	2.302673e-08	*																																																																																
		Effect	W	p	p<.05																																																																																																																		
3		Traits	0.3564105	4.864813e-18	*																																																																																																																		
4		EthnieVisages:Traits	0.3564105	4.864813e-18	*																																																																																																																		
7		Traits:Lunettes	0.5834814	2.302673e-08	*																																																																																																																		
8		EthnieVisages:Traits:Lunettes	0.5834814	2.302673e-08	*																																																																																																																		
Test de Mauchly (sphéricité)																																																																																																																							
<table border="1"> <thead> <tr> <th></th> <th></th> <th>Effect</th> <th>GGe</th> <th>p[GG]</th> <th>p[GG]&lt;.05</th> <th>HFe</th> <th>p[HF]</th> <th>p[HF]&lt;.05</th> </tr> </thead> <tbody> <tr> <td>3</td> <td></td> <td>Traits</td> <td>0.7100426</td> <td>8.162773e-49</td> <td>*</td> <td>0.7327269</td> <td>2.761505e-50</td> <td>*</td> </tr> <tr> <td>4</td> <td></td> <td>EthnieVisages:Traits</td> <td>0.7100426</td> <td>5.554754e-03</td> <td>*</td> <td>0.7327269</td> <td>5.056248e-03</td> <td>*</td> </tr> <tr> <td>7</td> <td></td> <td>Traits:Lunettes</td> <td>0.7904625</td> <td>5.361795e-34</td> <td>*</td> <td>0.8189768</td> <td>3.969476e-35</td> <td>*</td> </tr> <tr> <td>8</td> <td></td> <td>EthnieVisages:Traits:Lunettes</td> <td>0.7904625</td> <td>3.092782e-04</td> <td>*</td> <td>0.8189768</td> <td>2.535647e-04</td> <td>*</td> </tr> </tbody> </table>												Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]	p[HF]<.05	3		Traits	0.7100426	8.162773e-49	*	0.7327269	2.761505e-50	*	4		EthnieVisages:Traits	0.7100426	5.554754e-03	*	0.7327269	5.056248e-03	*	7		Traits:Lunettes	0.7904625	5.361795e-34	*	0.8189768	3.969476e-35	*	8		EthnieVisages:Traits:Lunettes	0.7904625	3.092782e-04	*	0.8189768	2.535647e-04	*																																																																	
		Effect	GGe	p[GG]	p[GG]<.05	HFe	p[HF]	p[HF]<.05																																																																																																															
3		Traits	0.7100426	8.162773e-49	*	0.7327269	2.761505e-50	*																																																																																																															
4		EthnieVisages:Traits	0.7100426	5.554754e-03	*	0.7327269	5.056248e-03	*																																																																																																															
7		Traits:Lunettes	0.7904625	5.361795e-34	*	0.8189768	3.969476e-35	*																																																																																																															
8		EthnieVisages:Traits:Lunettes	0.7904625	3.092782e-04	*	0.8189768	2.535647e-04	*																																																																																																															
Correction de sphéricité (Greenhouse-Geisser)																																																																																																																							

Le tableau ci-dessus donne les résultats de l'analyse, dont les effets principaux et d'interaction, le test de Mauchly (sphéricité) et la correction de la sphéricité en cas de non-respect de la condition d'application. En vert, les degrés de liberté (respectivement, du numérateur et du dénominateur) de l'indice  $F$ . En rouge, la valeur de l'indice  $F$ . En orange, la valeur  $p$ . Finalement, en bleu, les tailles d'effet générales par l'indice  $\eta_g^2$ .

Concernant le **test de sphéricité de Mauchly**, celui-ci **doit être non significatif** pour considérer l'égalité des covariances des erreurs du ou des facteur(s) intrasujet(s). Ce test n'est exécuté que pour les facteurs intrasujets comportant plus de deux modalités (ici, représenté par les effets 3, 4, 7 et 8 qui impliquent la variable « Traits »). Dans notre exemple, les sphéricités ne sont pas respectées. Pour cette raison, nous devons nous référer à la correction de Greenhouse-Geisser.

**Les comparaisons *a posteriori* ou tests post-hoc** permettent de savoir où se situent les différences significatives dans nos effets à plus de deux modalités. Il est nécessaire de recréer une variable d'ANOVA (sans inclure l'erreur des effets intrasujets) pour ensuite lancer les analyses post-hoc. La fonction `PostHocTest` comprend divers paramètres dont voici l'ordre :

- **Les facteurs à traiter** sous `which=`, en l'occurrence `which="Traits"`. Ceci signifie que nous ne souhaitons que les tests post-hoc de la variable « Traits », et non de l'effet d'interaction. Si vous souhaitez la totalité des variables, inscrivez `which=NULL`.
- **Le test post-hoc désiré** sous `method=`, en l'occurrence `method="hsd"`, représentant le test de Tukey HSD. Il est, entre autres, possible d'utiliser le test de Bonferroni sous "bonferroni" et le test de LSD sous "lsd".

```
ANOVAmixt_post <- aov(ScoreJugement~Traits*Lunettes*EthnieVisages,
                         data=Database_lg)
PostHocTest(ANOVAmixt_post, which="Traits",
            method="hsd", conf.level=.95)
```

```
Posthoc multiple comparisons of means : Tukey HSD
95% family-wise confidence level

$`Traits`
    diff      lwr.ci     upr.ci   pval
Conf-Attrait  1.0000000  0.75942377 1.2405762 < 2e-16 ***
Honnet-Attrait  0.9012945  0.66071827 1.1418707 < 2e-16 ***
Intell-Attrait  1.4097896  1.16921342 1.6503659 < 2e-16 ***
Succes-Attrait  1.2762945  1.03571827 1.5168707 < 2e-16 ***
Honnet-Conf   -0.0987055 -0.33928173 0.1418707 0.79538
Intell-Conf    0.4097896  0.16921342 0.6503659 3.6e-05 ***
Succes-Conf    0.2762945  0.03571827 0.5168707 0.01503 *
Intell-Honnet  0.5084951  0.26791892 0.7490714 1.0e-07 ***
Succes-Honnet  0.3750000  0.13442377 0.6155762 0.00022 ***
Succes-Intell -0.1334951 -0.37407137 0.1070811 0.55202

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Comme nous avons plusieurs VIs, les tailles d'effet individuelles ne permettent pas de connaître le pourcentage de la variance de la VD expliquée par l'*ensemble* de nos VIs. Pour le savoir, il peut être utile de calculer les **R<sup>2</sup>** (variance expliquée dans l'échantillon) ou **R<sup>2</sup> ajusté** (pourcentage de variance ajusté à la population).

```
summary(lm(ANOVAmixt_post))
```

```
Residual standard error: 0.8887 on 1010 degrees of freedom
Multiple R-squared:  0.3117, Adjusted R-squared:  0.2987
F-statistic: 24.07 on 19 and 1010 DF,  p-value: < 2.2e-16
```

Pour obtenir **les statistiques descriptives** en fonction de modalités désirées, la fonction `ezStats` se trouve plus simple d'utilisation que la fonction `group_by` lorsque vous effectuez une ANOVA à mesures répétées ou mixte, car elle se base sur les commandes de `ezANOVA`. Pour ce faire, précisez les facteurs désirés. Pour chaque effet principal, n'entrez qu'un facteur dans l'un des paramètres (`within=` ou `between=`). Pour chaque effet d'interaction, entrez les

facteurs désirés dans les paramètres (within= et/ou between= ; comme ci-dessous `Traits`, `Lunettes`). Il est toujours nécessaire d'inscrire la variable d'identification.

```
ezStats(data = Database_lg,
        dv = .(ScoreJugement),
        wid = .(Numparticipant),
        within = .(Traits, Lunettes),
        between = .(EthnieVisages))
```

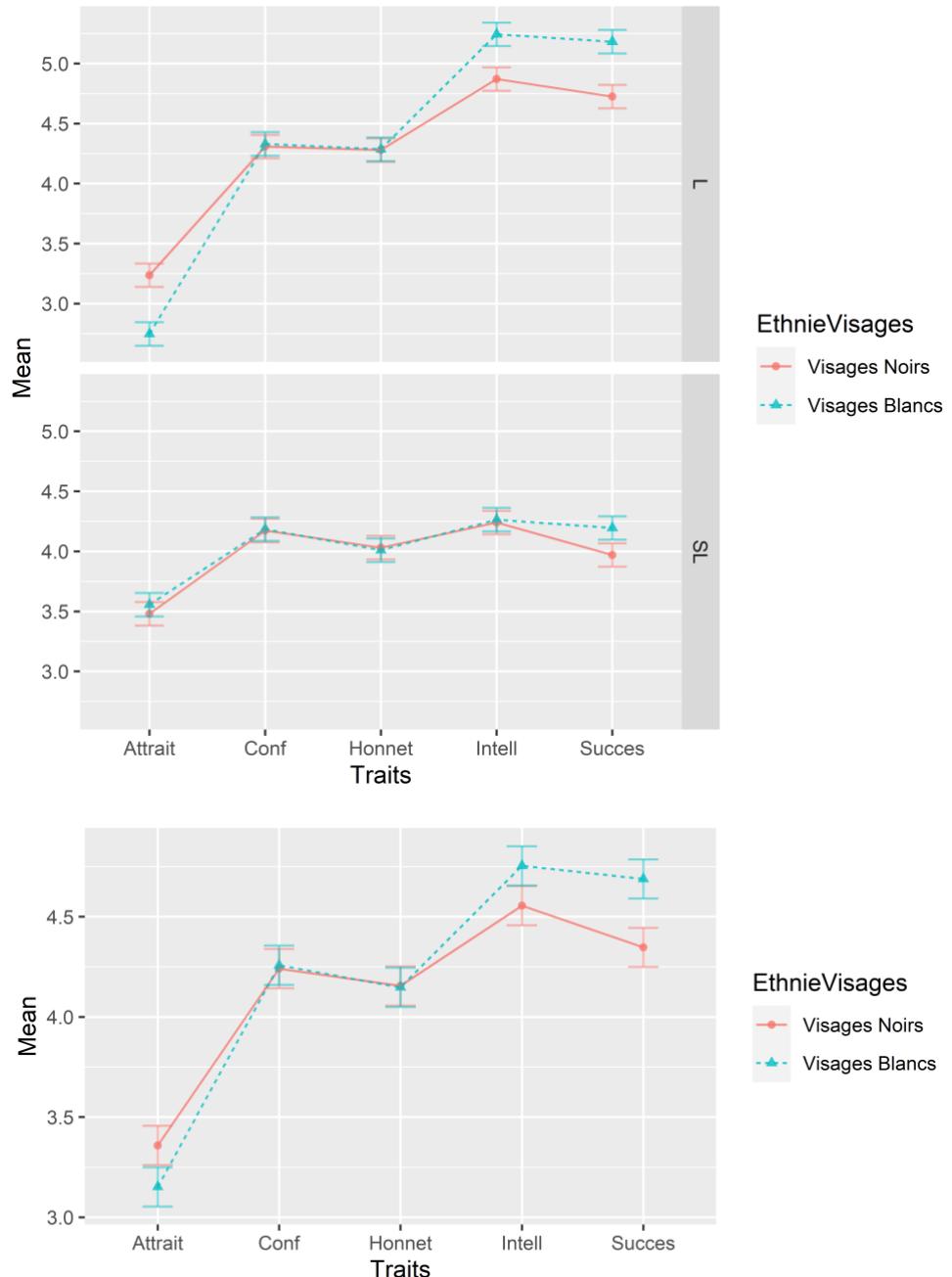
	EthnieVisages	Traits	Lunettes	N	Mean	SD	FLSD
1	Visages	Noirs	Attrait	L 49	3.236395	1.1631080	0.1959648
2	Visages	Noirs	Attrait	SL 49	3.479592	1.2235075	0.1959648
3	Visages	Noirs	Conf	L 49	4.307823	0.8067288	0.1959648
4	Visages	Noirs	Conf	SL 49	4.173469	0.8723234	0.1959648
5	Visages	Noirs	Honnet	L 49	4.278912	0.8917950	0.1959648
6	Visages	Noirs	Honnet	SL 49	4.030612	1.0478558	0.1959648
7	Visages	Noirs	Intell	L 49	4.870748	0.9565293	0.1959648
8	Visages	Noirs	Intell	SL 49	4.239796	0.8823489	0.1959648
9	Visages	Noirs	Succes	L 49	4.724490	0.8789929	0.1959648
10	Visages	Noirs	Succes	SL 49	3.969388	0.9137729	0.1959648
11	Visages	Blancs	Attrait	L 54	2.746914	0.8498309	0.1959648
12	Visages	Blancs	Attrait	SL 54	3.555556	0.8462830	0.1959648
13	Visages	Blancs	Conf	L 54	4.330247	0.8434858	0.1959648
14	Visages	Blancs	Conf	SL 54	4.185185	0.7863595	0.1959648
15	Visages	Blancs	Honnet	L 54	4.285494	0.8873384	0.1959648
16	Visages	Blancs	Honnet	SL 54	4.009259	0.7087109	0.1959648
17	Visages	Blancs	Intell	L 54	5.243827	0.9008574	0.1959648
18	Visages	Blancs	Intell	SL 54	4.263889	0.7044613	0.1959648
19	Visages	Blancs	Succes	L 54	5.182099	0.7733226	0.1959648
20	Visages	Blancs	Succes	SL 54	4.194444	0.7230482	0.1959648

Pour savoir comment se présente **graphiquement l'effet d'interaction**, la fonction `ezPlot` est simple d'utilisation. Suivant la même logique que les fonctions `ezANOVA` et `ezStats`, vous pouvez préciser l'interaction désirée. La fonction `ezPlot` comprend plusieurs paramètres supplémentaires dont voici l'ordre :

- **Le facteur présenté en abscisse (axe x)** sous `x=`, en l'occurrence `x=.(Traits)`. Ce facteur peut être interverti avec les autres paramètres.
- **Le facteur présenté en légendes (séparation en couleurs)** sous `split`, en l'occurrence `split=.(EthnieVisages)`. Ce facteur peut être interverti avec les autres paramètres.
- **Le(s) facteur(s) scindant le graphique en lignes et/ou en colonnes (si besoin)** sous `row` et/ou `col`, respectivement, en l'occurrence `row=.(Lunettes)`. Ce paramètre est nécessaire si vous souhaitez explorer graphiquement un effet d'interaction de trois ou quatre facteurs.

```
ezPlot(data = Database_lg,
       dv = .(ScoreJugement),
       wid = .(Numparticipant),
       within = .(Traits, Lunettes),
       between = .(EthnieVisages),
       x = .(Traits), split = .(EthnieVisages), row = .(Lunettes),
       do_lines = TRUE)
```

L'effet d'interaction entre les traits, le port de lunettes et l'ethnie des visages est développé dans cette commande et illustré dans le premier graphique. L'effet d'interaction entre les traits et l'ethnie des visages est uniquement illustré dans le second graphique. Nous remarquons alors que l'effet d'interaction entre les traits et l'ethnie des visages n'est en réalité présent que pour les visages portant des lunettes (et non pour ceux n'en portant pas).



**Notation :** « Une ANOVA mixte 5 (Traits, facteur intrasujet : succès, intelligence, confiance, attrait et honnêteté) x 2 (Port de lunettes, facteur intrasujet : avec et sans) x 2 (Ethnie du visage, facteur intersujet : peau noire ou peau blanche) a été effectué afin d'évaluer l'effet de ces variables sur le jugement des visages.

Il existe un effet principal significatif du trait de jugement sur l'évaluation de visages,  $F(4, 404) = 120.57, p < .05, \eta_g^2 = .24$  (avec correction Greenhouse-Geisser). Selon le test post-hoc de Tukey HSD (significativité fixée à .05), l'évaluation de l'intelligence ( $M = 4.66, SD = .94$ ) ne diffère pas de l'évaluation du succès ( $M = 4.53, SD = .94$ ), ces dernières étant supérieures à l'évaluation de la confiance ( $M = 4.25, SD = .82$ ) égale à l'évaluation de l'honnêteté ( $M = 4.15, SD = .89$ ), ces dernières étant supérieures à l'évaluation de l'attrait ( $M = 3.24, SD = 1.07$ ).

Un effet principal significatif du port de lunettes sur l'évaluation de visages est observé,  $F(1, 101) = 26.13, p < .05, \eta_g^2 = .03$ . Les visages portant des lunettes ( $M = 4.32, SD = 1.17$ ) obtiennent une évaluation des traits plus élevée que les visages ne portant pas de lunettes ( $M = 4.01, SD = .91$ ).

Il existe un effet d'interaction significatif entre les traits et le port de lunettes sur le jugement de visages,  $F(4, 404) = 64.79, p < .05, \eta_g^2 = .08$  (avec correction Greenhouse-Geisser). L'effet principal du port de lunettes est inversé pour l'évaluation de l'attrait, inexistant pour celles de la confiance et de l'honnêteté, mais reste valable pour celles de l'intelligence et du succès.

Il existe un effet d'interaction significatif entre les traits et l'ethnie sur le jugement de visages,  $F(4, 404) = 4.41, p < .05, \eta_g^2 = .01$  (avec correction Greenhouse-Geisser). Les visages noirs sont perçus comme plus attrayants, mais moins intelligents et ayant moins de succès professionnel que les visages blancs (sans différences sur la confiance et l'honnêteté).

Il existe un effet d'interaction significatif entre les traits, le port de lunettes et l'ethnie sur le jugement des visages,  $F(4, 404) = 6.22, p < .05, \eta_g^2 < .01$  (avec correction Greenhouse-Geisser). L'effet d'interaction décrit précédemment n'est en réalité présent que pour les visages portant des lunettes (sauf en ce qui concerne le succès, où les visages blancs sont jugés plus favorablement que les visages noirs avec ou sans lunettes).

Aucun autre effet significatif n'a été trouvé. L'ensemble des variables et effets d'interaction expliquent 30% de la variance du jugement de visages dans la population,  $R_{adj}^2 = .30$ . »

#### 4.4.4.6 MANOVA (Multivariate Analysis of Variance)

Fonctions et packages nécessaires : Anova, leveneTest {car} ; boxM, etasq {heplots} ; manova, summary.aov, TukeyHSD {stats} ; etaSquared {lsr} ; describe {psych}

La MANOVA permet de tester l'existence de différences significatives d'un ou plusieurs facteurs (VI intersujet(s)) sur plusieurs variables quantitatives différentes (VD). Contrairement à l'ANOVA mixte, **on ne s'intéresse pas aux différences entre VDs** (ou aux effets de facteur(s) intrasujet(s)). La MANOVA permet ainsi savoir s'il existe un effet des facteurs intersujets, à la fois globalement sur les variables quantitatives (VD) considérées *dans leur ensemble* et spécifiquement sur chacune des variables quantitatives prises *une à une*.

Exemple : Nous souhaitons savoir s'il existe une différence entre filles et garçons adolescent·es (facteur intersujet `Sexe`) sur l'évaluation de l'importance de cinq types de problèmes, soit les *problèmes familiaux* (VD 1, `pb_fam`), les *problèmes scolaires* (VD 2, `pb_sco`), les *problèmes médicaux* (VD 3, `pb_med`), les *problèmes sociaux* (VD 4, `pb_soc`), les *problèmes légaux* (VD 5, `pb_leg`), variables tirées de la base de données `Database`. Ces problèmes sont codés par un psychologue sur une échelle de dix niveau (0 = *absence de problèmes* à 9 = *problèmes très graves*). **Nous ne nous intéressons pas aux différences de moyennes entre problèmes**, mais cherchons à savoir s'il existe une différence globale entre filles et garçons (sur les VD considérées dans leur ensemble) et où se situe exactement cette différence éventuelle (sur quelle VD spécifiquement). Dans cet exemple, il n'y a qu'une VI, mais comme pour les ANOVA précédentes, on peut en ajouter d'autres (avec les effets d'interaction qui s'ajouteront).

**Attention** : Pour que les commandes suivantes soient exécutées correctement, il est nécessaire que la ou les VI soient bel et bien considérées comme des facteurs par R (cf. chapitre 4.2.7). De plus, afin de faciliter la lecture des tableaux, renommez les modalités de vos facteurs avec le label approprié (cf. chapitre 4.2.8).

Dans les conditions d'application du test, **l'homogénéité des variances** de nos différents groupes sur toutes nos variables quantitatives doit être vérifiée grâce au **test de Levene** (ici un seul exemple pour `pb_med`).

```
leveneTest(pb_med~Sexe, data=Database, center=mean) #Ex.: pb_med
```

Nous devons vérifier **l'homogénéité de la matrice de variances/covariances grâce au test de Box**, devant être non significatif pour considérer la condition d'application comme respectée.

```
boxM(cbind(pb_fam, pb_sco, pb_med, pb_soc, pb_leg) ~ Sexe,
      data=Database)
```

```
Box's M-test for Homogeneity of Covariance Matrices
```

```
data: Y
Chi-Sq (approx.) = 13.448, df = 15, p-value = 0.5677
```

Dans le cas présent, l'homogénéité de la matrice de variances/covariances peut être considérée respectée,  $\chi^2(15) = 13.45$ ,  $p > .05$ .

Les fonctions suivantes permettent d'**exécuter la MANOVA** et d'en observer les résultats par le biais des *effets globaux*, soit les effets des variables quantitatives considérées dans leur ensemble, et les *effets spécifiques*, soit les effets sur chaque variable quantitative considérée une à une. Concernant les effets globaux, sous `test=`, il est possible de spécifier d'autres tests, tels que le *Lambda de Wilks* par "Wilks", la *Trace de Hotelling* par "Hotelling-Lawley" ou la *Plus grande racine de Roy* par "Roy". Par défaut, la *Trace de Pillai* est sélectionnée (par "Pillai").

```

MANOVA_PbSx <- manova(cbind(pb_fam, pb_sco, pb_med, pb_soc, pb_leg)
                        ~ Sexe, data=Database)
summary(MANOVA_PbSx, test="Pillai") #Effets globaux
summary.aov(MANOVA_PbSx) #Effets individuels

```

	Df	Pillai	approx F	num Df	den Df	Pr(>F)
sexe	1	0.18572	4.2422	5	93	0.001613 **
Residuals	97					
---						
Signif. codes:	0	***	0.001	**	0.01	*, 0.05 ., 0.1 , , 1

### Effets globaux

*L'effet du facteur intersujet est testé pour l'ensemble des variables quantitatives*

Response pb_fam :	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sexe	1	56.29	56.286	12.505	0.0006244 ***
Residuals	97	436.62	4.501		
---					
Signif. codes:	0	***	0.001	**, 0.01	*, 0.05 ., 0.1 , , 1
Response pb_sco :	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sexe	1	5.19	5.1948	0.9622	0.3291
Residuals	97	523.71	5.3991		
---					
Response pb_med :	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sexe	1	8.33	8.3348	1.7434	0.1898
Residuals	97	463.75	4.7809		
---					
Response pb_soc :	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sexe	1	9.58	9.5848	2.3023	0.1324
Residuals	97	403.83	4.1632		
---					
Response pb_leg :	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sexe	1	2.49	2.4852	0.442	0.5077
Residuals	97	545.35	5.6222		

### Effets spécifiques

*L'effet du facteur intersujet est testé pour chacune des variables quantitatives séparément*

Sur le premier tableau (effets globaux), nous pouvons constater que le facteur `Sexe` a un effet global sur les VD considérées dans leur ensemble. Cela signifie que le sexe a un effet général sur les problèmes rencontrés par les adolescent·es. Cependant, le second tableau nous permet d'en savoir un peu plus sur cet effet et de savoir sur quel(s) problème(s) précisément les deux sexes se démarquent.

Sur ce second tableau (effets spécifiques), nous remarquons que les garçons et les filles ne se distinguent que sur les problèmes familiaux, et non sur les quatre autres problèmes où la différence n'est pas significative.

Un effet significatif peut être détecté grâce à des échantillons très importants, mais sans grands écarts dans les moyennes. La **taille d'effet** renseigne sur la grandeur des effets significatifs observés. Les tailles d'effets communément utilisées dans l'analyse de variance sont le  $\eta^2$  (éta-carré) et le  $\eta^2_{partiel}$ . Elles font référence à la force ou à la magnitude de l'association, soit la proportion de la variance totale de la VD expliquée par les variations de chaque VI. Les tailles d'effet peuvent être adaptées au test effectué pour les **effets globaux**.

```

MANOVA_eSizeGLO <- lm(MANOVA_PbSx)
etasq(Anova(MANOVA_eSizeGLO), anova=TRUE) #SI Pillai (par defaut)
## OU
etasq(MANOVA_eSizeGLO, test="Wilks") #SINON autres tests (ex:Wilks)

```

```

> etasq(Anova(MANOVA_eSizeGLO), anova=TRUE)

Type II MANOVA Tests: Pillai test statistic
  eta^2 Df test stat approx F num Df den Df Pr(>F)
sex 0.18572 1 0.18572 4.2422 5 93 0.001613 **
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> etasq(MANOVA_effectSize,test="Wilks")
  eta^2
sex 0.1857187

```

Pour ce qui est des **effets spécifiques**, il est nécessaire de calculer les valeurs une à une comme décrit ci-dessous. Voici deux exemples parmi les cinq à effectuer.

```

etasquared(aov(pb_fam~sexe, data=Database)) #Problemes Familiaux
etasquared(aov(pb_sco~sexe, data=Database)) #Problemes Scolaires

```

```

> etaSquared(aov(pb_fam~sexe, data=DProblemes)) #Problemes familiaux
  eta.sq eta.sq.part
sexe 0.1141916 0.1141916
> etaSquared(aov(pb_sco~sexe, data=DProblemes)) #Problemes scolaires
  eta.sq eta.sq.part
sexe 0.009821736 0.009821736

```

Pour les **tests *a posteriori*** (ou **tests post-hoc**), ils ne sont pas nécessaires si le(s) facteur(s) intersujet(s) ne contiennent pas plus de deux modalités, ce qui est le cas dans notre exemple. Dans la situation où un facteur intersujet contient plus de deux modalités, voici les commandes à exécuter. La fonction n'est pas adaptée aux analyses multivariées, c'est pourquoi il est nécessaire de lancer plusieurs post-hoc consécutifs comme pour les tailles d'effets individuels (ceci, uniquement pour les effets principaux et d'interaction significatifs).

```
TukeyHSD(aov(pb_med~psy), data=Database) #Problemes Medicaux
```

```
$`factor(psy)`:
    diff      lwr      upr     p adj
2-1 1.5708333 0.4179120 2.723755 0.0045902
3-1 1.7761905 0.3667445 3.185636 0.0095482
3-2 0.2053571 -1.0907112 1.501425 0.9246368
```

Dans cette situation, nous pouvons identifier deux différences à l'intérieur du facteur intersujet sur les problèmes médicaux, soit les modalités 2 et 3 ne différant pas entre elles, mais étant plus élevées que la modalité 1.

Pour demander **toutes les moyennes et écarts-types** en fonction des différentes modalités du (ou des) facteur(s) intersujet(s) et des variables dépendantes (ici, spécifiées par leur colonne, de la 7<sup>ème</sup> à la 11<sup>ème</sup>, dans la base de données), procédez distinctement ainsi :

```

#POUR 1 facteur inter :
by(Database[, 7:11], Database$Sexe, describe)
#POUR 2+ fact inter :
by(Database[, 7:11], list(Database$Sexe, Database$Psy), describe)

```

sex: Garçons													
	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
pb_med	1	63	2.95	2.27	2	2.80	1.48	0	8	8	0.47	-0.91	0.29
pb_sco	2	63	4.52	2.33	4	4.55	2.97	0	9	9	-0.06	-0.82	0.29
pb_soc	3	63	4.60	2.04	5	4.59	2.97	1	9	8	0.02	-0.82	0.26
pb_fam	4	63	4.79	2.24	5	4.76	2.97	0	9	9	0.14	-0.75	0.28
pb_leg	5	63	4.08	2.36	4	3.94	2.97	0	9	9	0.43	-0.66	0.30

sex: Filles													
	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
pb_med	1	36	3.56	2.03	3.0	3.43	1.48	1	8	7	0.52	-0.99	0.34
pb_sco	2	36	5.00	2.31	5.0	5.00	2.97	1	9	8	0.01	-0.99	0.38
pb_soc	3	36	5.25	2.03	5.0	5.17	2.22	2	9	7	0.30	-1.05	0.34
pb_fam	4	36	6.36	1.88	6.5	6.43	2.22	3	9	6	-0.26	-1.25	0.31
pb_leg	5	36	3.75	2.38	3.5	3.57	2.22	0	9	9	0.55	-0.50	0.40

**Notation :** « Une MANOVA à 1 facteur intersujet (Sexe : garçon ou fille) a été effectuée afin d'évaluer l'importance de cinq types de problèmes (familiaux, scolaires, médicaux, sociaux et légaux) rencontrés par des adolescent·es filles ou garçons. L'indice de la Trace de Pillai indique que le sexe des adolescent·es interrogé·es a une influence globale sur les variables considérées dans leur ensemble,  $V = .19$ ,  $F(5, 93) = 4.24$ ,  $p < .05$ ,  $\eta^2 = .19$ . Les filles indiquent rencontrer globalement davantage de problèmes que les garçons. La différence est notamment significative sur l'importance des problèmes familiaux,  $F(1, 97) = 12.51$ ,  $p < .05$ ,  $\eta^2 = .11$ . Les filles ( $M = 6.36$ ,  $SD = 1.88$ ) indiquent rencontrer davantage de problèmes familiaux que les garçons ( $M = 4.79$ ,  $SD = 2.24$ ). Cependant, le sexe des adolescent·es interrogé·es n'a pas d'influence sur l'importance des problèmes scolaires,  $F(1, 97) = .96$ ,  $p > .05$ ,  $\eta^2 = .01$ , des problèmes médicaux,  $F(1, 97) = 1.74$ ,  $p > .05$ ,  $\eta^2 = .02$ , des problèmes sociaux,  $F(1, 97) = 2.30$ ,  $p > .05$ ,  $\eta^2 = .02$ , et des problèmes légaux,  $F(1, 97) = .44$ ,  $p > .05$ ,  $\eta^2 < .01$ .

#### 4.4.4.7 Contrastes orthogonaux (*Comparaisons a priori*)

Fonctions et packages nécessaires : `lm {stats}` ; `summarize {dplyr}`

Dans les ANOVA développées aux chapitres précédents, rejeter  $H_0$  signifie qu'il existe au moins une différence entre les moyennes. Quand le facteur (intersujet ou intrasujet) ne contient que deux modalités, il suffit de calculer les moyennes pour savoir de quelle différence il s'agit. Néanmoins, quand le facteur contient plus de deux modalités, rejeter  $H_0$  ne nous dit pas où se situent exactement ces différences entre moyennes. Nous connaissons déjà la possibilité des comparaisons post-hoc quand nous n'avons pas d'hypothèses préalables (*a posteriori*) grâce aux tests de Tukey ou de Bonferroni. **Lorsque nous avons des hypothèses préalables (*a priori*)**, nous pouvons définir des **contrastes**.

Exemple : Lors d'une ANOVA à un facteur intersujet à propos des différences de racisme symbolique de la part de footballeurs, arbitres et fans de football, nous pouvons émettre l'hypothèse sur la base des recherches antérieures que les fans (étant plus jeunes) obtiendront un score de racisme symbolique inférieur aux deux autres (*contraste 1*) et que ces derniers, soit les footballeurs et arbitres, ne différeraient pas entre eux sur ce score de racisme symbolique (*contraste 2*). Ce test est plus précis (et donc plus puissant) que les post-hoc qui testent toutes les différences possibles.

La première étape est de **définir les contrastes** par le biais de l'attribution des poids à chaque groupe comparé. Pour le premier contraste de l'exemple, nous comparons le premier groupe aux deux autres : ainsi, le poids à attribuer au premier doit être deux fois plus élevé qu'aux deux autres. Pour le deuxième contraste, nous comparons un groupe à un autre (ainsi, les poids sont identiques). Les commandes suivantes permettent d'illustrer ces propos.

```
Cont1 <- c(1/2, 1/2, -1) #Contraste 1: (Foot, Arbitres) vs. Fans
Cont2 <- c( 1, -1,  0) #Contraste 2: Foot vs. Arbitres
```

Pour le premier contraste, si nous additionnons chaque poids ( $\frac{1}{2} + \frac{1}{2} - 1$ ), nous obtenons une somme nulle. Pour le deuxième contraste, la somme est également nulle. Pour que les contrastes soient orthogonaux, la somme des produits doit être égale à 0, ce qui est le cas.

La prochaine étape est la **création de la matrice de contraste**. Sous `constant=`, la valeur insérée doit être  $\frac{1}{j}$  où  $j$  est égal au nombre de groupes (en l'occurrence,  $\frac{1}{3}$ ).

```
mat.temp <- rbind(constant=1/3, Cont1, Cont2) ; mat.temp
mat <- solve(mat.temp) ; mat #Inversion de la matrice temporaire
mat <- mat[ , -1] ; mat #Finalisation de la matrice de contraste
```

```
> mat
      Cont1 Cont2
[1,] 0.3333333 0.5
[2,] 0.3333333 -0.5
[3,] -0.6666667 0.0
```

Après avoir lancé les commandes précédentes, la matrice de contraste est prête à être utilisée.

```
ModelContr <- lm(RacismeSymb ~ Group, data = DFootRacisme,
                   contrasts=list(Group = mat))
summary(ModelContr)
```

```
Call:
lm(formula = racismesymb ~ grp, data = DFootRacisme, contrasts =
  list(grp = mat))

Residuals:
    Min      1Q  Median      3Q     Max 
-1.88889 -0.52713 -0.02713  0.47965  1.97287 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 2.72201   0.08743 31.133 < 2e-16 ***
grpCont1    0.70801   0.18581  3.810 0.000277 ***  
grpCont2   -0.36176   0.21377 -1.692 0.094636 .    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7129 on 77 degrees of freedom
(2 observations deleted due to missingness)
Multiple R-squared:  0.1626,    Adjusted R-squared:  0.1409 
F-statistic: 7.478 on 2 and 77 DF,  p-value: 0.001077
```

Dans le tableau ci-dessus, nous pouvons à nouveau confirmer l'effet principal du groupe à la ligne « F-statistic » où nous retrouvons les valeurs du  $F$ , du  $p$  ainsi que du  $R^2$  identiques. En rouge, nous constatons que nos hypothèses *a priori* ont été confirmées : seul le contraste 1 est significatif. Ainsi, les footballeurs et les arbitres (réunis) obtiennent un score de racisme symbolique significativement supérieur aux fans (*contraste 1*), les footballeurs et arbitres ne se distinguent pas (*contraste 2*). Nous pouvons **contrôler que le poids des contrastes** effectués par l'analyse a été correctement implémenté dans l'analyse.

```
attributes(ModelContr$qr$qqr) $contrasts #Contrôle des contrastes
```

```
$grp
      Cont1 Cont2
Footballeurs 0.3333333 0.5
Arbitres     0.3333333 -0.5
Fans         -0.6666667 0.0
```

**Rappel :** Pour obtenir les moyennes et écart-types en fonction des modalités, référez-vous au chapitre *Statistiques descriptives* (4.3.10).

```
# A tibble: 3 x 3
  grp        Moyenne EcartType
  <fct>      <dbl>    <dbl>
1 Footballeurs  2.78    0.731
2 Arbitres      3.14    0.822
3 Fans          2.25    0.588
```

**Notation :** « Une ANOVA à 1 facteur intersujet (Groupe : footballeurs, arbitres ou fans) a été effectuée afin d'évaluer les différences de score de racisme symbolique en fonction de ces groupes.

Il existe un effet principal du groupe sur le score de racisme symbolique,  $F(2, 77) = 7.48$ ,  $p < .05$ ,  $R^2_{ajusté} = .14$ . Une analyse de contrastes orthogonaux a révélé que conformément aux hypothèses, les fans ( $M = 2.25$ ,  $SD = .59$ ) obtiennent un score significativement plus bas que les footballeurs ( $M = 2.78$ ,  $SD = .73$ ) et les arbitres ( $M = 3.14$ ,  $SD = .82$ ), ces derniers ne différant pas entre eux.

#### 4.4.5 Tests non paramétriques

Lorsque les conditions d'application ne sont pas respectées, notamment dans le cas d'une distribution non normale (test de Kolmogorov significatif, histogrammes avec asymétrie positive (effet plancher, courbe en i) ou avec asymétrie négative (effet plafond, courbe en j), boxplots avec moustaches inégales, médiane non centrée et valeurs extrêmes), des groupes qui ne sont pas équivalents en nombre ou des variances non homogènes, il est possible d'utiliser des tests non paramétriques (sans condition d'application) transformant les variables quantitatives en variables ordinaires (rangs). Ainsi, le but de ces tests est identique à l'ANOVA ou au t-test, mais sur des variables ordinaires. (Pour une question de simplicité, nous reprendrons les bases de données utilisées précédemment bien qu'elles respectent les conditions d'application des tests paramétriques.)

##### 4.4.5.1 *U de Mann-Whitney/W de Wilcoxon : 2 groupes intersujets ou indépendants*

Fonction et package nécessaires : wilcox.test {stats}

La statistique du test *U* (identique au test de la somme des rangs de Wilcoxon) réunit les deux échantillons et ordonne les observations par ordre croissant de taille. Le test calcule le nombre de fois où un résultat du groupe 1 précède un résultat du groupe 2, ainsi que le nombre de fois où un résultat du groupe 2 précède un résultat du groupe 1. Contrairement au *t* ou au *F*, l'indice *U* ou *W* est d'autant plus petit que les groupes sont différents : en somme, plus *U* ou *W* est petit, plus la différence est élevée.

Exemple : Nous souhaitons tester l'existence d'une différence dans le jugement du succès pour des visages avec lunettes (`SuccesL`) en fonction de l'ethnie des visages présentés (`EthnieVisages`), variables tirées de la base de données `Database`. Rappelons que les participant·es n'ont eu à juger qu'une seule ethnie des visages (groupes *indépendants*). Nous avons donc **1 variable nominale à 2 modalités** (soit des visages de peau noire, soit des visages de peau blanche) et **1 variable comme ordinaire** (le jugement de succès de visages avec lunettes transformé en rangs).

```
wilcox.test(SuccesL ~ EthnieVisages, data = Database,
            paired=FALSE, alternative="two.sided")
```

```
Wilcoxon rank sum test with continuity correction

data: Database$SuccesL by Database$EthnieVisages
W = 918, p-value = 0.007241
alternative hypothesis: true location shift is not equal to 0
```

Notez que la fonction, comme pour le t-test, vous permet de spécifier l'hypothèse alternative grâce à `alternative` (pour plus d'explications, voir le chapitre 4.4.3). Le résultat vous rappelle l'hypothèse alternative que vous avez spécifié. Dans le cas présent, nous avons testé une hypothèse bilatérale (différence entre les deux groupes, sans préciser dans quel sens). La fonction utilise un *W* pour évaluer le *U*, mais les deux tests sont équivalents (même valeur *p*).

Les rangs moyens à partir desquels *U* et *W* sont calculés ne sont jamais mentionnés comme mesure de tendance centrale. Il est donc nécessaire de faire usage des **médianes** ainsi que l'indice de dispersion qui correspond, **l'intervalle interquartile** ou **IQR**, pour écrire votre phrase de conclusion.

**Rappel :** Pour obtenir les médianes et écarts interquartiles (ou *IQR*) en fonction des facteurs, référez-vous au chapitre *Statistiques descriptives* (4.3.10).

# A tibble: 2 x 4	EthnieVisages	Nombre	Mediane	IQR
	<fct>	<int>	<dbl>	<dbl>
1	Visages Noirs	49	4.75	1
2	Visages Blancs	54	5.25	0.938

**Notation :** « Les visages avec lunettes de peau noire ( $Mdn = 4.75$ ,  $IQR = 1.00$ ) diffèrent significativement des visages avec lunettes de peau blanche ( $Mdn = 5.25$ ,  $IQR = .94$ ) quant au jugement de succès qui leur a été posé,  $W = 918$ ,  $p < .05$ . »

#### 4.4.5.2 Kruskal-Wallis : n groupes intersujets ou indépendants

Fonctions et packages nécessaires : `kruskal.test {stats}` ; `kruskalmc {pgirmess}`

Le test de Kruskal-Wallis permet de tester la différence des scores de plus de deux groupes indépendants, (alors que le *U* de Mann-Whitney et le *W* de Wilcoxon ne permettent de comparer que deux groupes indépendants).

Exemple : Nous souhaitons tester l'existence d'une différence du score de racisme symbolique ([RacismeSymb](#)) entre un groupe d'arbitres, un groupe de fans et un groupe de joueurs de football ([Groupe](#)), variables tirées de la base de données [Database](#). Nous avons donc **1 variable nominale à 3 modalités indépendantes** (soit le groupe d'arbitres, le groupe de fans ou le groupe de joueurs) et **1 variable considérée comme ordinaire** (le score de racisme symbolique transformé en rangs).

```
kruskal.test(RacismeSymb ~ Groupe, data=Database)
```

```
Kruskal-Wallis rank sum test
data: racismesymb by groupe
Kruskal-Wallis chi-squared = 13.491, df = 2, p-value = 0.001176
```

Les rangs moyens à partir desquels le test est calculé ne sont jamais mentionnés comme mesure de tendance centrale. Il est donc nécessaire de faire usage des **médianes** ainsi que l'indice de dispersion qui correspond, **l'intervalle interquartile** ou **IQR**, pour écrire votre phrase de conclusion.

**Rappel :** Pour obtenir les médianes et écarts interquartiles (ou *IQR*) en fonction des modalités, référez-vous au chapitre *Statistiques descriptives* (4.3.10).

# A tibble: 3 x 4	groupe	Nombre	Mediane	IQR
	<fct>	<int>	<dbl>	<dbl>
1	Footballeurs	43	2.75	1
2	Arbitres	17	3.25	1.12
3	Fans	22	2.12	0.688

**Remarque :** Nous savons qu'il existe un effet principal du groupe sur le score de racisme symbolique, mais ne connaissons pas où se situent ces différences. Il est possible que les différences se situent uniquement entre les arbitres et les fans, mais nous ne pouvons le savoir sans le tester : c'est pourquoi, comme à chaque fois que nous testons plus de deux groupes, un **test post-hoc** est utile. Dans le cas où le test de Kruskal-Wallis est non significatif, il n'y a aucune utilité d'effectuer un test post-hoc.

```
kruskalmc(Racismesymb~Groupe, data=Database)
```

```
Multiple comparison test after Kruskal-Wallis
p.value: 0.05
Comparisons
      obs.dif critical.dif difference
Footballeurs-Arbitres 12.03178    16.68213 FALSE
Footballeurs-Fans     15.60307    14.58238 TRUE
Arbitres-Fans         27.63485    18.62779 TRUE
```

La colonne « difference » permet de savoir si la différence est significative ("TRUE") ou non ("FALSE") à hauteur de  $p < .05$ . Ainsi, nous voyons que les différences sont toutes significatives, hormis pour la comparaison « Footballeurs-Arbitres ».

**Notation :** « Les groupes diffèrent significativement sur le score de racisme symbolique,  $\chi^2(2) = 13.49$ ,  $p < .05$ . Plus précisément, des tests post-hoc ont montré que la différence est significative entre le groupe de footballeurs ( $Mdn = 2.75$ ,  $IQR = 1.00$ ) et celui des fans ( $Mdn = 2.12$ ,  $IQR = .69$ ), et entre le groupe d'arbitres ( $Mdn = 3.25$ ,  $IQR = 1.12$ ) et celui des fans,  $p < .05$ . Aucune différence n'a été soulevée entre le groupe de footballeurs et celui des arbitres,  $p > .05$ . »

#### 4.4.5.3 Wilcoxon : 2 facteurs intrasujets ou groupes appariés : wilcox.test {stats}

Le test du signe de Wilcoxon permet de tester la différence entre deux mesures répétées, transformées en rangs.

Exemple : Nous souhaitons tester l'existence d'un jugement d'intelligence pour des visages avec lunettes (`IntellL`) plus élevé que pour des visages sans lunettes (`IntellSL`), variables tirées de la base de données `Database`. Rappelons que les participant·es ont eu à juger à la fois des visages avec lunettes et des visages sans lunettes (mesures appariées). Nous pouvons considérer que nos 2 variables quantitatives sont regroupées en **1 variable nominale à 2 modalités appariées** (avec ou sans la présence de lunettes) et **1 variable ordinaire** (le jugement de l'intelligence transformé en rangs).

```
wilcox.test(Database$IntellL, Database$IntellSL,
            paired=TRUE, alternative="greater")
```

Notez que la fonction, comme pour le t-test, vous permet de spécifier l'hypothèse alternative grâce à `alternative`. Le résultat vous rappelle l'hypothèse alternative que vous avez spécifiée. Dans le cas présent, nous avons testé une hypothèse alternative unilatérale `greater` (`IntellSL` plus élevé que `IntellL`).

```
Wilcoxon signed rank test with continuity correction
data: Database$IntellL and Database$IntellSL
V = 3721, p-value = 9.339e-13
alternative hypothesis: true location shift is greater than 0
```

Les rangs moyens ne sont jamais mentionnés comme mesure de tendance centrale. Il est donc nécessaire de faire usage des **médiennes** ainsi que l'indice de dispersion qui correspond, **l'invervalle inter-quartile ou IQR**, pour écrire votre phrase de conclusion.

```
# A tibble: 1 x 5
  Nombre Mediane_IntellSL IQR_IntellSL Mediane_IntellL IQR_IntellL
  <int>        <dbl>       <dbl>        <dbl>       <dbl>
1     103        4.25        1             5          1.25
```

**Notation :** « Les visages avec lunettes ( $Mdn = 5.00$ ,  $IQR = 1.25$ ) sont jugés plus intelligents que les visages sans lunettes ( $Mdn = 4.25$ ,  $IQR = 1.00$ ),  $V = 3721$ ,  $p < .05$ . »

#### 4.4.5.4 ANOVA de Friedman : n facteurs intrasujets ou groupes appariés

Fonctions et packages nécessaires : `friedman.test {stats}` ; `friedmanmc {pgirmess}`

La commande de l'ANOVA de Friedman demande de convertir notre base de données en matrice ou en vecteurs contenant seulement les variables utilisées dans la fonction. De plus, elle ne fonctionne pas s'il y a des valeurs manquantes. Premièrement, nous allons créer une **DataFrame** à partir des variables que nous utiliserons. Deuxièmement, nous préciserons qu'il faut supprimer les valeurs manquantes. Finalement, nous compilerons la fonction `friedman.test` en précisant que nous utilisons une matrice avec la fonction `as.matrix`.

Exemple : Nous souhaitons tester l'existence d'une différence entre les trois variables de jugement suivantes : le succès (`SuccesL`), l'honnêteté (`HonnetL`) et l'attrait (`AttraitL`) des visages avec lunettes, variables tirées de la base de données `Database`. Nous pouvons considérer que nous avons **1 variable nominale à 3 modalités appariées** (le succès, l'honnêteté et l'attrait) et **1 variable ordinaire** (le jugement des visages avec lunettes en rangs).

```
DataFrameAHS <- cbind(Database$AttraitL,
                      Database$HonnetL,
                      Database$SuccesL)

friedman.test(as.matrix(na.omit(DataFrameAHS)))
```

```
Friedman rank sum test

data: as.matrix(DataFrameAHS)
Friedman chi-squared = 146, df = 2, p-value < 2.2e-16
```

**Remarque** : Nous savons qu'il existe un effet principal de la catégorie sur le score de jugement des visages avec lunettes, mais ne connaissons pas *où* se situent ces différences. Pour savoir entre quelles modalités les différences sont significatives, un **test post-hoc** est à utiliser. Dans le cas où l'ANOVA n'est pas significative, il n'y a aucune utilité d'effectuer un tel test.

```
friedmanmc(as.matrix(na.omit(DataFrameAHS)))
```

La colonne « difference » permet de savoir si la différence est significative ("TRUE") ou non ("FALSE") à hauteur de  $p < .05$ . Nous voyons que les différences sont toutes significatives.

Comparisons			
	obs.dif	critical.dif	difference
1-2	107	34.36007	TRUE
1-3	166	34.36007	TRUE
2-3	59	34.36007	TRUE

Les rangs moyens utilisés pour le calcul du test ne sont jamais mentionnés comme mesure de tendance centrale. Il est nécessaire de faire usage des **médianes** ainsi que l'indice de dispersion qui correspond, **l'invervalle inter-quartile** ou **IQR**, pour écrire votre phrase de conclusion.

```
> median(AttraitL) ; IQR(AttraitL)
[1] 3
[1] 1
> median(HonnetL) ; IQR(HonnetL)
[1] 4.25
[1] 1
> median(SuccesL) ; IQR(SuccesL)
[1] 5
[1] 1.083333
```

**Notation** : « Les visages avec lunettes sont jugés avec significativement plus de succès ( $Mdn = 5.00$ ,  $IQR = 1.08$ ) que d'honnêteté ( $Mdn = 4.25$ ,  $IQR = 1.00$ ) que d'attrait ( $Mdn = 3.00$ ,  $IQR = 1.00$ ),  $\chi^2(2) = 146$ ,  $p < .05$ . »

#### 4.4.6 Corrélations bivariées : cor.test {stats} ; rcorr {Hmisc}

L'analyse de corrélation permet de tester l'existence d'une relation positive ou négative (ou nulle) entre deux variables quantitative ( $r$  de Bravais-Pearson) ou deux variables ordinaires ( $\rho$  de Spearman,  $\rho$ ). Plusieurs paramètres du test de corrélation désiré sont disponibles :

- **Hypothèse alternative** : Le paramètre `alternative=` permet de spécifier l'hypothèse alternative. Voici les possibilités :
  - **Hypothèse bilatérale** (`alternative="two.sided"`) : L'hypothèse alternative est bilatérale. Nous ne nous avançons pas quant à la direction de la relation (elle peut être positive ou négative).
  - **Hypothèse unilatérale, lien positif** (`alternative="greater"`) : L'hypothèse alternative est unilatérale et suppose une relation *positive* entre les deux variables.
  - **Hypothèse unilatérale, lien négatif** (`alternative="less"`) : L'hypothèse alternative est unilatérale et suppose une relation *négative* entre les deux variables.
- **Méthode de corrélation** : Le caractère `method=` permet de spécifier le type de corrélation désirée. Il est possible d'en spécifier plusieurs à la fois afin de comparer les résultats (voir la commande ci-dessous pour des **variables données**). Voici deux des trois possibilités<sup>1</sup> :
  - **Corrélation de Bravais-Pearson** (`method="pearson"`) : La corrélation spécifiée est celle de Bravais-Pearson pour des variables quantitatives.
  - **Corrélation de Bravais-Pearson** (`method="spearman"`) : La corrélation spécifiée est celle de Spearman pour des variables ordinaires (ou des variables quantitatives ne suivant pas une distribution normale incluant des valeurs extrêmes).

```
cor.test(VAR1, VAR2, method=c("pearson", "spearman"))
```

- **Intervalle de confiance** : Le paramètre `conf.level=` permet de spécifier l'intervalle de confiance (p. ex.: 90% ou 99%), soit la valeur de  $p$  utilisée pour tester la significativité ou non de notre relation. Par défaut, et de manière générale, l'intervalle de confiance utilisé est 95%.

```
cor.test(VAR1, VAR2, conf.level=.99) #La valeur de p doit être <.01
```

Concernant la force de la relation, nous pouvons utiliser les critères suivants : une corrélation est faible dès lors que  $r < .30$ , moyenne pour un  $r$  de  $.30$  à  $.50$ , forte dès lors que  $r > .50$ .

---

<sup>1</sup> La troisième possibilité est le *tau* de Kendall ("kendall"). Elle est également calculée sur des rangs, mais la formule sous-jacente est différente du  $\rho$  de Spearman. La statistique *tau* représente une différence de probabilité entre la possibilité que les deux variables soient du même ordre contre la possibilité que les deux variables ne soient pas du même ordre.

#### 4.4.6.1 Corrélation de Bravais-Pearson

Dans le cas où nous souhaitons tester l'existence d'une relation entre deux variables **quantitatives** et lorsque les conditions d'application le permettent, nous pouvons utiliser la corrélation de Bravais-Pearson. Nous pouvons également tester la corrélation de plusieurs paires de variables quantitatives à la fois à l'aide d'une seule commande (Situation 2).

Situation 1 : Nous souhaitons tester l'existence d'un lien positif entre le nombre de problèmes familiaux (`pb_fam`) et le nombre de problèmes scolaires (`pb_sco`) rencontrés par des adolescent·es, variables tirées de la base de données `Database`. Nous souhaitons donc tester la relation positive de **2 variables quantitatives**.

```
cor.test(Database$pb_fam, Database$pb_sco,  
         alternative="greater", method="pearson")
```

```
Pearson's product-moment correlation  
  
data: pb_fam and pb_sco  
t = 6.2605, df = 97, p-value = 5.228e-09  
alternative hypothesis: true correlation is greater than 0  
95 percent confidence interval:  
 0.4063957 1.0000000  
sample estimates:  
 cor  
 0.536454
```

Notez que nous avons spécifié une hypothèse alternative unilatérale, en l'occurrence une relation positive entre nos deux variables. Si votre base de données contient des valeurs manquantes, le test sera tout de même effectué sans utiliser la paire d'observations concernée par la valeur manquante. Les degrés de liberté notés entre () sont donnés par le « *df* » (ou *degrees of freedom*). Les degrés de liberté peuvent être calculés manuellement :  $df = n - 2$ , où  $n = \text{nombre d'observations}$ . En l'occurrence :  $r(99-2) = r(97)$ .

**Notation :** « Il existe une relation positive forte entre les problèmes familiaux et les problèmes scolaires,  $r(97) = .54$ ,  $p < .05$ . »

Situation 2 : Nous souhaitons tester l'existence d'un lien entre les différents types de problèmes, soit familiaux (`pb_fam`), scolaires (`pb_sco`), sociaux (`pb_soc`), légaux (`pb_leg`) et médicaux (`pb_med`), rencontrés par des adolescent·es, variables tirées de la base de données `Database`. Nous souhaitons donc tester la relation de **plusieurs variables quantitatives, paire par paire** en une seule commande.

```
rcorr(as.matrix(Database[,c("pb_fam", "pb_sco", "pb_soc",  
"pb_leg", "pb_med")]), type="pearson")
```

pb_fam	1.00	0.54	0.67	0.18	0.15
pb_sco	0.54	1.00	0.57	0.35	0.14
pb_soc	0.67	0.57	1.00	0.30	0.17
pb_leg	0.18	0.35	0.30	1.00	0.67
pb_med	0.15	0.14	0.17	0.67	1.00

n=	99
----	----

P	pb_fam	pb_sco	pb_soc	pb_leg	pb_med
pb_fam	0.0000	0.0000	0.0688	0.1414	
pb_sco	0.0000		0.0000	0.0004	0.1719
pb_soc	0.0000	0.0000		0.0023	0.0941
pb_leg	0.0688	0.0004	0.0023		0.0000
pb_med	0.1414	0.1719	0.0941	0.0000	

**Remarque :** Nous ne nous intéressons qu'aux résultats en-dessus (ou en-dessous, à votre convenance) de la diagonale puisque le tableau est symétrique.

Sur ce tableau, nous pouvons connaître les différentes **valeurs de  $r$** , mais également le **nombre d'observations** présentes dans ces différentes corrélations (ici, pour toutes  $n = 99$ ) et les **valeurs  $p$** . Pour la notation, vous pouvez vous référer à celle de l'exemple précédent en indiquant les différentes valeurs de  $r(n-2)$  significatives.

Si votre base de données contient des valeurs manquantes, les tests seront tout de même effectués sans utiliser la paire d'observations concernée par la valeur manquante (en anglais, *excluding cases pairwise*). Pour l'illustrer, voici un exemple avec des données fictives :

> x <- c(3,4,2,5,7,4,7,4,6,NA,5,3)
> y <- c(6,7,4,4,7,6,8,3,9,2,5,NA)
> z <- c(NA,8,2,3,1,4,7,1,3,4,NA,5)
> rcorr(as.matrix(XYZ[, c("x","y","z")]), type="pearson")
x    y    z
x 1.00 0.59 0.17
y 0.59 1.00 0.31
z 0.17 0.31 1.00

n	x	y	z
x	11	10	8
y	10	11	8
z	8	8	9

P	x	y	z
x	0.0707	0.6857	
y	0.0707		0.4558
z	0.6857	0.4558	

#### 4.4.6.2 Corrélation de Spearman

Dans le cas où nous souhaitons tester l'existence d'une relation entre deux variables **ordinales** ou lorsque la normalité de variables quantitatives n'est pas respectée, nous pouvons utiliser la corrélation de Spearman, qui comme les autres tests non paramétriques transforme les scores en rangs.

Les commandes de la corrélation de Spearman sont identiques à celle de la corrélation de Bravais-Pearson, à l'exception que nous précisons `method="spearman"` à la place de `method="pearson"`. Pour des informations détaillées sur la procédure ou les différents éléments de la console R, référez-vous au chapitre précédent.

Situation 1 : Nous souhaitons tester l'existence d'un lien positif entre le nombre de problèmes familiaux (`pb_fam`) et le nombre de problèmes scolaires (`pb_sco`) rencontrés par des adolescent·es, variables tirées de la base de données `Database`. Supposons que ces variables ne suivent pas une distribution tout à fait normale.

```
cor.test(Database$pb_fam, Database$pb_sco,
          alternative="greater", method="spearman")
```

```
Spearman's rank correlation rho

data: pb_fam and pb_sco
S = 73704, p-value = 2.909e-09
alternative hypothesis: true rho is greater than 0
sample estimates:
rho
0.5441932
```

Le  $\rho$  (rho) de Spearman, noté parfois  $r_s$ , est identique au  $r$  de Bravais-Pearson. Finalement, il semblerait que la normalité n'ait pas été problématique pour notre test de corrélation.

**Notation :** « Il existe une relation positive forte entre les problèmes familiaux et les problèmes scolaires,  $r_s(97) = .54, p < .05$ . »

Situation 2 : Nous souhaitons tester l'existence d'un lien entre les différents types de problèmes, soit familiaux (`pb_fam`), scolaires (`pb_sco`), sociaux (`pb_soc`), légaux (`pb_leg`) et médicaux (`pb_med`), rencontrés par des adolescent·es, variables tirées de la base de données `Database`. Supposons que ces variables ne suivent pas une distribution normale. Nous souhaitons tester la relation de **plusieurs variables, paire par paire** en une seule commande, en les transformant en rangs.

```
rcorr(as.matrix(Database[,c("pb_fam", "pb_sco", "pb_soc",
                           "pb_leg", "pb_med")]), type="spearman")
```

```
      pb_fam pb_sco pb_soc pb_leg pb_med
pb_fam  1.00  0.54  0.67  0.19  0.17
pb_sco  0.54  1.00  0.58  0.33  0.16
pb_soc  0.67  0.58  1.00  0.27  0.18
pb_leg  0.19  0.33  0.27  1.00  0.65
pb_med  0.17  0.16  0.18  0.65  1.00

n= 99

P
      pb_fam pb_sco pb_soc pb_leg pb_med
pb_fam          0.0000 0.0000 0.0607 0.0907
pb_sco 0.0000          0.0000 0.0009 0.1067
pb_soc 0.0000 0.0000          0.0068 0.0742
pb_leg 0.0607 0.0009 0.0068          0.0000
pb_med 0.0907 0.1067 0.0742 0.0000
```

Pour la notation, vous pouvez vous référer à celle de l'exemple précédent en indiquant les différentes valeurs de  $r_s$  significatives.

#### 4.4.6.3 Corrélation partielle

Fonction et package nécessaires : pcor.test {ppcor}

La corrélation partielle est calculée lorsque nous souhaitons mettre en évidence l'association entre deux variables ( $x$  et  $y$ ) en contrôlant (en partialisant) l'influence d'une autre variable ( $z$ ).

Exemple : Nous souhaitons connaître la corrélation partielle entre les tracas quotidiens (`TracasQ`) et les problèmes de santé (`PbSante`), en partialisant (contrôlant) le nombre d'événements critiques (`EveCrit`), variables tirées de la base de données `Database`.

Autrement dit : quel lien reste-t-il entre les tracas quotidiens et les problèmes de santé si nous retirons l'influence des événements critiques sur ces deux variables ? Ou encore, y a-t-il un effet médiateur des événements critiques sur le lien entre les tracas quotidiens et les problèmes de santé ?

**Attention :** Cette analyse ne fonctionne que lorsque nous n'avons pas de valeurs manquantes. Ainsi, il peut être utile de regarder si nous avons des valeurs manquantes avant de débuter l'analyse. La fonction `is.na()` permet de connaître pour quelle(s) valeur(s) il est vrai (ou `TRUE`) qu'elle est manquante.

```
is.na(Database$TracasQ)
is.na(Database$PbSante)
is.na(Database$EveCrit)
```

```
> is.na(Database$TracasQ)
[1] FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
> is.na(Database$PbSante)
[1] TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE
> is.na(Database$EveCrit)
[1] FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

En l'occurrence, nous voyons que nos variables contiennent un total de quatre valeurs manquantes et que, par conséquent, une analyse de corrélation partielle n'est pas encore possible. Si tel est le cas, suivez la procédure ci-dessous. Si vos variables ne contiennent pas de valeurs manquantes, suivez directement la commande d'analyse.

**En cas de valeurs manquantes :** Nous pouvons nous baser sur le chapitre *Supprimer les lignes avec valeurs manquantes de toutes variables confondues* (4.2.12.1). Pour commencer, nous créerons un double de notre base de données (cela est surtout utile si vous comptez effectuer d'autres analyses avec d'autres variables). Puis, nous exécuterons la commande supprimant toutes les lignes avec valeurs manquantes de nos trois variables (*excluding cases listwise*).

```
Database_noNA <- Database
Database_noNA <- Database_noNA[complete.cases(Database_noNA[, 
c("TracasQ", "PbSante", "EveCrit")]), ]
```

Il est possible de contrôler que la commande a été exécutée correctement en utilisant la commande `is.na()` décrite ci-dessus. Si tout a fonctionné, il sera désormais noté `FALSE` sur toutes les valeurs.

**Analyse :** Les deux variables liées sont nommées  $x$  (soit, `TracasQ`) et  $y$  (soit, `PbSante`), la variable médiatrice (à contrôler) est nommée  $z$  (soit, `EveCrit`). Les variables  $x$  et  $y$  peuvent être interverties car l'influence de la variable médiatrice  $z$  est retirée des deux variables  $x$  et  $y$ .

Cependant,  $z$  doit rester la variable médiateuse. Nous utiliserons notre base de données sans valeurs manquantes (soit, **Database noNA**).

Les degrés de liberté sont égaux à  $n - v$ , où  $n$  le nombre de participant·es et  $v$  le nombre de variables, dans notre cas  $17 - 3$ . Pour savoir si l'effet médiateur est fort ou non, il vous faut comparer la corrélation partielle avec la corrélation simple développée aux chapitres précédents. Cependant, pour une comparaison correcte, vous devez utiliser la base de données sans les valeurs manquantes.

```
cor.test(x = Database_noNA$TracasQ, y = Database_noNA$PbSante)
pcor.test(x = Database_noNA$TracasQ,
          y = Database_noNA$PbSante,
          z = Database_noNA$EveCrit)
```

```
estimate      p.value statistic n gp Method
1 0.7466037 0.0008920932 4.199105 17 1 pearson
```

**Notation :** « La corrélation simple entre les tracas quotidiens et les problèmes de santé est élevée,  $r(15) = .83, p < .05$ . Comme la corrélation partielle entre les tracas quotidiens et les problèmes de santé en contrôlant l'influence des événements critiques,  $r_p(14) = .75, p < .05$ , est presque aussi élevée, nous concluons une absence de médiation du lien entre les tracas quotidiens et les problèmes de santé par le nombre d'événements critiques. »

#### **4.4.6.4 Corrélation semi-partielle**

## *Fonctions et packages nécessaires : spcor.test {ppcor}*

La corrélation semi-partielle est calculée lorsque nous souhaitons mettre en évidence l'influence spécifique d'une variable  $x$  sur une variable  $y$  en retirant l'influence d'une variable  $z$  sur cette variable  $y$ . Ainsi, contrairement à la corrélation partielle, nous retirerons l'influence de cette tierce variable uniquement sur la seconde variable de base  $y$  et non sur la première  $x$ .

Exemple : Nous souhaitons connaître l'influence spécifique des tracas quotidiens ([TracasQ](#)) sur la santé physique ([PbSante](#)) en soustrayant l'influence des événements critiques ([EveCrit](#)) de la santé physique, variables tirées de la base de données [Database](#). Autrement dit : quelle part de la variance de la santé physique, indépendamment des événements critiques, peut être expliquée par les tracas quotidiens ?

**Attention :** Cette analyse ne fonctionne que lorsque nous n'avons pas de valeurs manquantes. Ainsi, il peut être utile de regarder si nous avons des valeurs manquantes avant de débuter l'analyse. La fonction `is.na()` permet de connaître pour quelle(s) valeur(s) il est vrai (ou `TRUE`) qu'elle est manquante.

```
is.na(Database$TracasQ)  
is.na(Database$PbSante)  
is.na(Database$EveCrit)
```

```
> is.na(Database$TracasQ)
[1] FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE FALSE
> is.na(Database$PbSante)
[1] TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE TRUE
> is.na(Database$EveCrit)
[1] FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[12] FALSE FALSE
```

En l'occurrence, nous voyons que nos variables contiennent un total de quatre valeurs manquantes et que, par conséquent, une analyse de corrélation semi-partielle n'est pas encore possible. Si tel est le cas, suivez la procédure ci-dessous. Si vos variables ne contiennent pas de valeurs manquantes, suivez directement la commande d'analyse.

**En cas de valeurs manquantes :** Nous pouvons nous baser sur le chapitre *Supprimer les lignes avec valeurs manquantes de toutes variables confondues* (4.2.12.1). Pour commencer, nous créerons un double de notre base de données (cela est surtout utile si vous comptez effectuer d'autres analyses avec d'autres variables). Puis, nous exécuterons la commande supprimant toutes les lignes avec valeurs manquantes de nos trois variables. (*excluding cases listwise*).

```
Database_noNA <- Database
Database_noNA <- Database_noNA[complete.cases(Database_noNA[, c("TracasQ", "PbSante", "EveCrit")]),]
```

Il est tout à fait possible de contrôler que la commande a été exécutée correctement en utilisant la commande `is.na()` décrite ci-dessus. Si tout a fonctionné, il sera désormais noté `FALSE` sur toutes les valeurs.

**Analyse :** Les deux variables liées sont nommées  $x$  (soit, `TracasQ`) et  $y$  (soit, `PbSante`), la variable à contrôler est nommée  $z$  (soit, `EveCrit`). Les variables  $x$  et  $y$  ne doivent pas être interverties, ainsi que  $z$  devant rester la variable contrôle. Les variables  $x$  et  $y$  doivent maintenir un ordre spécifique, car nous retirerons l'influence de la variable contrôle *uniquement* de  $y$  et non de  $x$ , d'où corrélation *semi*-partielle, contrairement à la corrélation partielle où l'influence de la variable à contrôler est retirée des deux ( $x$  et  $y$ ). Nous utiliserons notre base de données sans valeurs manquantes (soit, `Database_noNA`).

Les degrés de liberté sont égaux à  $n - v$ , où  $n$  le nombre de participant·es et  $v$  le nombre de variables, dans notre cas  $17 - 3$ . Pour savoir si le lien spécifique est fort, il vous faut comparer la corrélation semi-partielle avec la corrélation simple développée aux chapitres précédents. Cependant, pour une comparaison correcte, vous devez utiliser la base de données sans les valeurs manquantes.

```
cor.test(x = Database_noNA$TracasQ, y = Database_noNA$PbSante)
spcor.test(x = Database_noNA$TracasQ,
            y = Database_noNA$PbSante,
            z = Database_noNA$EveCrit)
```

```
estimate      p.value statistic   n gp Method
1 0.5224529  0.03787675  2.292615 17  1 pearson
```

**Notation :** « Les tracas quotidiens ont un lien spécifique (non lié aux événements critiques vécus) fort avec les problèmes de santé, dans le sens suivant : plus le nombre de tracas quotidiens est important, plus les problèmes de santé sont nombreux, indépendamment des événements critiques vécus,  $r_{sp}(14) = .52, p < .05$  . »

#### 4.4.6.5 Représentation graphique des coefficients de corrélation

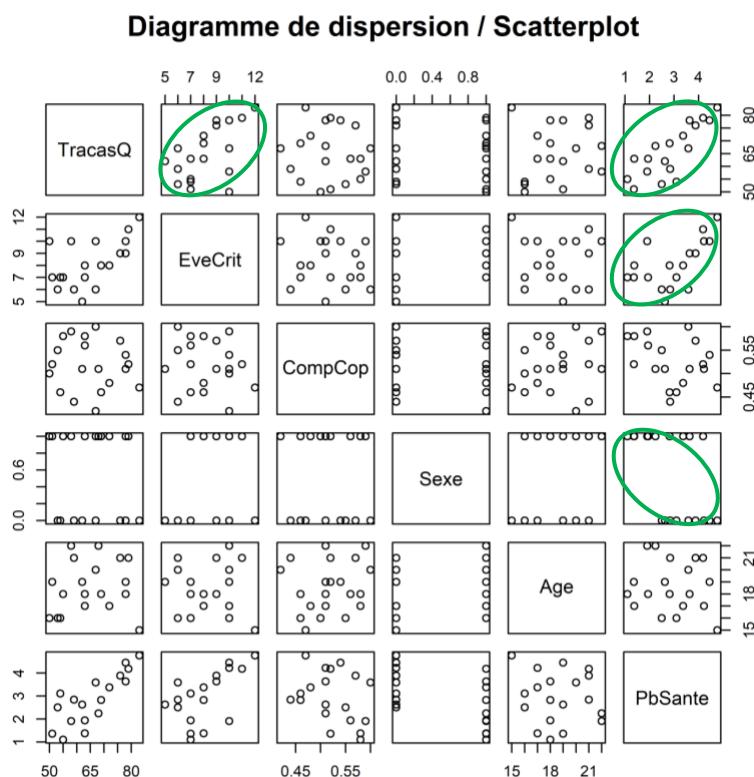
##### 4.4.6.5.1 Diagramme de dispersion ou Scatterplot

###### Fonctions et packages nécessaires : pairs {graphics}

Avant de démarrer des analyses corrélationnelles avec plusieurs variables quantitatives, il est bon d'explorer graphiquement les relations entre ces variables au travers d'un diagramme de dispersion (ou *Scatterplot*). Si vous entrez plus de deux variables, cela crée une matrice de diagrammes de dispersion avec les variables prises 2 à 2 (redondante au-dessous et au-dessous de la diagonale).

Exemple : Nous souhaitons obtenir un diagramme de dispersion pour les variables quantitatives suivantes : tracas quotidiens (), événements critiques (), compétence de coping (), sexe (), âge et problèmes de santé (), variables tirées de la base de données **Database**.

```
pairs(~TracasQ + EveCrit + CompCop + Sexe + Age + PbSante,
      data = Database,
      main = "Diagramme de dispersion / Scatterplot")
```



D'après le diagramme de dispersion, il semble que les tracas quotidiens sont positivement corrélés aux événements critiques ainsi qu'aux problèmes de santé. Les événements critiques, quant à eux, semblent positivement liés aux problèmes de santé. Le sexe semble être corrélé aux problèmes de santé. La présence d'une corrélation avec une variable dichotomique se remarque visuellement par un décalage entre les dispersions des deux groupes.

#### 4.4.6.5.2 Corrélogrammes

Fonctions et packages nécessaires : corrgram {corrgram}

Les corrélogrammes permettent de visualiser graphiquement les coefficients de corrélation.

Exemple : Nous souhaitons observer, sous forme graphique, les coefficients de corrélation calculés à l'Exemple 2 du chapitre *Corrélation de Bravais-Pearson* (4.4.6.1).

Tout d'abord, **sélectionnez les variables** à partir desquelles les coefficients de corrélation seront calculés. Il y a deux possibilités de sélectionner les variables désirées : premièrement, en insérant le numéro des colonnes en question (p. ex.: `c(12, 13, 14, 15, 16)` ou simplement `c(12:16)`), ou directement les noms des variables comme présenté ci-dessous.

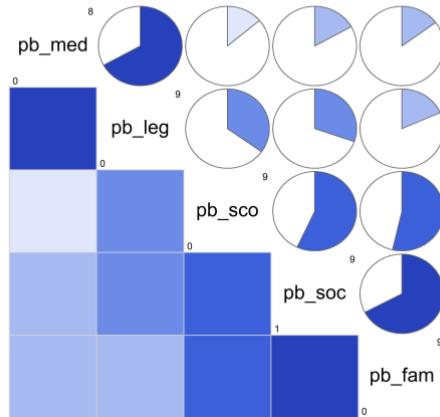
```
DCorrGram <- Database[,c("pb_fam", "pb_sco", "pb_soc", "pb_leg",  
"pb_med")]
```

Grâce à la base de données créée à partir des variables d'intérêt, nous pouvons représenter sous forme **graphique** (en l'occurrence, sous forme d'une matrice de graphiques) nos coefficients de corrélation. Plusieurs combinaisons sont possibles :

- **Type de graphique** : Les paramètres `lower.panel=` et `upper.panel=` permettent de spécifier le graphique désiré, respectivement, en-dessous et en-dessus de la diagonale. Il est possible de ne demander qu'un seul (en spécifiant le panel désiré, "lower" ou "upper"). La couleur des graphiques permet de connaître également le sens de la corrélation (le rouge représente une corrélation négative, et le bleu une corrélation positive). Plus la couleur est foncée, plus la corrélation est élevée. Les options sont :
  - **Intensité de couleurs** : Sous `panel.fill`, l'intensité de la couleur des figures permet de connaître la force de la corrélation.
  - **Barres de corrélation** : Sous `panel.bar`, l'intensité de la couleur et la hauteur des barres permettent de connaître la force de la corrélation.
  - **Diagrammes circulaires** : Sous `panel.pie`, la proportion de remplissage des diagrammes circulaires permet de connaître la force de la corrélation.
  - **Nuages de points** : Sous `panel.pts`, les nuages de points des paires de variables sont représentés.
  - **Coefficients de corrélation avec IC** : Sous `panel.conf`, les coefficients de corrélations sont représentés avec leurs intervalles de confiance (IC) à 95%.
- **Eléments de la diagonale** : La diagonale séparant les deux types de graphiques peut contenir diverses informations, dont notamment :
  - **Nom des variables** : Le paramètre `text.panel=panel.txt` permet d'y insérer le nom des variables.
  - **Minimum et maximum** : Le paramètre `diag.panel=panel.minmax` permet d'y insérer les minimum et maximum des variables.
  - **Courbe de densité** : Le paramètre `diag.panel=panel.density` permet d'obtenir la courbe de distribution des variables.
- **Ordre des variables (ACP)** : Le paramètre `order=` permet d'ordonner les variables en utilisant une analyse en composantes principales (si `order=TRUE`), ou de maintenir l'ordre inséré tel que sur la commande précédente (si `order=FALSE`).

```
corrgram(DCorrGram, order=TRUE, lower.panel=panel.shade,
         upper.panel=panel.pie, text.panel=panel.txt,
         diag.panel=panel.minmax, cor.method="pearson",
         main="Correlogramme des problemes rencontres")
```

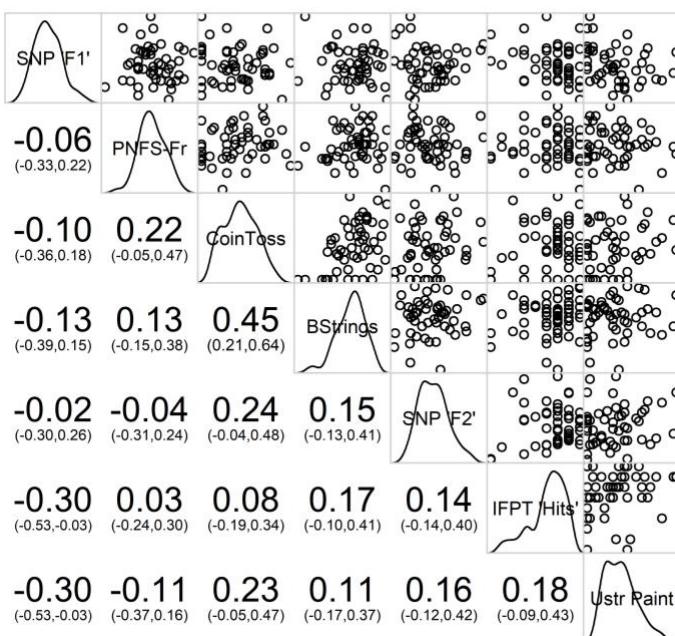
Correlogramme des problemes rencontres par les jeunes



Voici un autre exemple de corrélogramme avec corrélations négatives dont l'ordre a été décidé par une analyse en composantes principales (ou ACP). Nous pouvons constater que les liens négatifs et positifs ont été regroupés suggérant probablement deux composantes principales.

```
corrgram(DCorrGram, order=TRUE, lower.panel=panel.conf,
         upper.panel=panel.pts, text.panel=panel.txt,
         diag.panel=panel.density, cor.method="pearson",
         main="Correlogramme d'Illusory Pattern Perception")
```

Correlogramme d'Illusory Pattern Perception



#### 4.4.7 Indice de consistance interne – Alpha de Cronbach : alpha {psych}

L'indice de consistance interne (homogénéité, cohérence interne, fiabilité) donné par l' $\alpha$  de Cronbach se base sur le calcul de la corrélation moyenne entre différents items d'un test censé mesurer la même notion. Si nos items sont suffisamment corrélés positivement, nous pouvons alors créer un seul score moyen de ces items.

Exemple : Un questionnaire est composé de 10 items d'une échelle censée mesurer le degré d'émotions négatives. Nous souhaitons savoir si les participant·es perçoivent ces différents items comme relevant de la même dimension (toutes plus ou moins positivement corrélées) ou si elles sont indépendantes les unes des autres (corrélations nulles). De plus, nous souhaitons nous assurer que les items sont toutes codés dans le même sens ("0" étant un degré bas d'émotions négatives et "5" un degré élevé).

**Tout d'abord**, si notre base de données contient des variables qui ne sont pas toutes en rapport avec notre mesure d'émotions négatives, nous devons rassembler les items d'émotions négatives sous une seule et même **base de données annexe** en précisant leur position dans la **base de données originale**.

```
negativeEmotions <- Database[, c(6:15)]
```

Ici, les 10 items d'émotions négatives se situent de la 6<sup>ème</sup> colonne à la 15<sup>ème</sup> colonne de notre base de données originale.

**Cependant**, nous ne savons toujours pas si tous les items sont codés dans le même sens ou non. La fonction `alpha()` nous permet d'utiliser un paramètre qui vérifie directement quels items sont codés dans le sens inverse grâce à `check.keys=TRUE` pour calculer l'indice  $\alpha$  final (en somme, comme si tous les items avaient dès le départ été codés dans le même sens).

```
psych::alpha(negativeEmotions, check.keys=TRUE)
```

Sur le tableau ci-dessous, vous obtenez l' $\alpha$  (brut) de nos 10 items mesurant le degré d'émotions négatives. La valeur  $\alpha$  est égale à .76 et est donc satisfaisante ( $\alpha > .70$ ).

Reliability analysis						
Call: alpha(x = negativeEmotions, check.keys = TRUE)						
raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase mean	sd median_r
0.76	0.75	0.77	0.23	3.1	0.019	2.1 0.84 0.21
lower alpha		upper		95% confidence boundaries		
0.73		0.76		0.8		

Item statistics							
	n	raw.r	std.r	r.cor	r.drop	mean	sd
E01	88	0.43	0.44	0.32	0.28	2.66	1.40
E02	88	0.60	0.59	0.53	0.46	1.61	1.59
E03-	88	0.59	0.60	0.54	0.47	3.26	1.30
E04-	88	0.34	0.35	0.22	0.17	1.83	1.50
E05	88	0.52	0.51	0.43	0.38	1.57	1.40
E06	88	0.58	0.58	0.51	0.44	2.33	1.49
E07	88	0.74	0.70	0.69	0.61	2.47	1.84
E08	88	0.77	0.76	0.76	0.66	2.00	1.82
E09-	88	0.69	0.70	0.68	0.59	2.73	1.31
E10-	88	0.30	0.35	0.23	0.19	0.95	0.99

**Warning message:**  
In `alpha(negativeEmotions, check.keys = TRUE)` :  
Some items were negatively correlated with total scale and were automatically reversed.  
This is indicated by a negative sign for the variable name.

Sur le tableau précédent, nous pouvons connaître la **contribution des différents items** à l'indice  $\alpha$ . En l'occurrence, celui qui est le plus corrélé à la mesure globale est l'item E08 (valeur de .77) et ceux qui le sont le moins sont les items E04 et E10 (valeurs de .34 et .30, respectivement).

Un **message d'avertissement** nous précise que certains items sont corrélés négativement avec la mesure globale et ont été automatiquement inversés (comme précisé précédemment avec le paramètre `check.keys=TRUE`) : ces items sont indiqués par un signe "-".

Reliability if an item is dropped:									
	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
E01	0.76	0.75	0.77	0.25	3.0	0.020	0.024	0.26	
E02	0.74	0.73	0.75	0.23	2.7	0.022	0.021	0.20	
E03-	0.74	0.73	0.75	0.23	2.7	0.021	0.024	0.20	
E04-	0.78	0.77	0.78	0.27	3.3	0.018	0.020	0.28	
E05	0.75	0.74	0.76	0.24	2.9	0.021	0.024	0.22	
E06	0.74	0.73	0.75	0.23	2.7	0.021	0.025	0.19	
E07	0.71	0.71	0.72	0.21	2.4	0.024	0.019	0.19	
E08	0.71	0.70	0.71	0.20	2.3	0.025	0.020	0.19	
E09-	0.72	0.71	0.73	0.21	2.4	0.023	0.021	0.19	
E10-	0.77	0.77	0.78	0.27	3.3	0.019	0.021	0.26	

Le tableau ci-dessus nous permet de savoir quel serait la valeur de l' $\alpha$  si l'item avait été supprimé de la mesure globale. En l'occurrence, nous observons qu'en retirant l'item E04, nous obtenons un  $\alpha$  à peine plus élevé (.78 au lieu de .76). Nous l'avions déjà remarqué par sa corrélation faible à la mesure globale. Pour une augmentation si faible, nous avons autant la possibilité de le garder que de le retirer.

Non missing response frequency for each item							
	0	1	2	3	4	5 miss	
E01	0.09	0.09	0.27	0.26	0.18	0.10	0.7
E02	0.34	0.20	0.18	0.11	0.09	0.07	0.7
E03	0.15	0.39	0.17	0.22	0.03	0.05	0.7
E04	0.05	0.12	0.15	0.23	0.20	0.25	0.7
E05	0.22	0.41	0.15	0.09	0.09	0.05	0.7
E06	0.11	0.22	0.24	0.18	0.16	0.09	0.7
E07	0.22	0.17	0.09	0.17	0.16	0.19	0.7
E08	0.34	0.11	0.14	0.14	0.16	0.11	0.7
E09	0.10	0.16	0.31	0.31	0.05	0.08	0.7
E10	0.01	0.01	0.06	0.11	0.45	0.35	0.7

Ce dernier tableau nous donne les statistiques descriptives des différentes valeurs des items sous forme de fréquences (ici, quels sont les pourcentages de réponses données par les participant·es égales à 0, 1, 2, 3, 4 et 5).

**Remarque :** Lorsque vous travaillez sur des échelles qui comportent plusieurs variables qui ont déjà été validées par d'autres études, il n'est nullement recommandé de retirer des variables de la mesure globale simplement pour « remonter » la valeur de l' $\alpha$  si la valeur est déjà au-dessus de .70 (au-dessous, cela peut s'envisager). Il se peut que le fait de retirer une variable modifie la signification du tout.

**Attention :** Nous ne pouvons pas calculer un score moyen des 10 items car, comme l'indique l'un des tableaux, certains items sont codés dans un sens inverse aux autres. Pour calculer un score global d'émotions négatives (pour chaque participant·e), il est nécessaire d'inverser les échelles concernées au travers d'un recodage (voir chapitre *Recoder des variables (4.2.9)*).

**Notation :** « Les dix items de l'échelle du degré d'émotions négatives ont une consistance interne très satisfaisante,  $\alpha$  de Cronbach = .76. »

#### 4.4.8 Régression linéaire

Fonctions et packages nécessaires : `lm {stats}` ; `ncvTest {car}` ; `allEffects {effects}` ; `stepAIC {MASS}` ; `lm.beta {lm.beta}`

La régression linéaire permet de savoir si une variable quantitative (régression simple) ou plusieurs variables quantitatives (régression multiple), appelées *prédicteurs* (ou VI quantitatives), sont possible(s) de prédire les valeurs d'une autre variable quantitative appelée *critère* (ou VD).

##### 4.4.8.1 Régression linéaire simple et multiple

Exemple : Nous souhaitons savoir quels types de problèmes, soit médicaux (`pb_med`), scolaires (`pb_sco`), sociaux (`pb_soc`) et familiaux (`pb_fam`), prédisent la consommation de drogue (`Conso_Dro`) chez des adolescent·es, variables tirées de la base de données `Database`. L'analyse à effectuer est une **régression linéaire multiple** avec les quatre types de problèmes comme **prédicteurs** et la consommation de drogue comme **critère**.

```
RegMult <- lm(Conso_Dro ~ pb_med + pb_sco + pb_soc + pb_fam,  
               data = Database)  
summary(RegMult)
```

```
Call:  
lm(formula = Conso_Dro ~ pb_med + pb_sco + pb_soc + pb_fam, data = Database)  
  
Residuals:  
    Min      1Q  Median      3Q     Max  
-2.58292 -0.96333  0.03475  0.97839  2.67773  
  
Coefficients:  
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 3.54821   0.39571  8.967 2.89e-14 ***  
pb_med       0.32400   0.06100  5.312 7.26e-07 ***  
pb_sco      -0.12888   0.07130 -1.808  0.07387 .  
pb_soc       0.29422   0.09221  3.191  0.00193 **  
pb_fam       0.09980   0.08209  1.216  0.22716  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 1.304 on 94 degrees of freedom  
Multiple R-squared:  0.3915,    Adjusted R-squared:  0.3656  
F-statistic: 15.12 on 4 and 94 DF,  p-value: 1.407e-09
```

La partie du tableau « Coefficients » traite des différents prédicteurs, des **coefficients de la pente** (coefficients B non standardisés, sous *Estimate*, en rouge) et de leur **significativité ou non** (t-tests en vert, rejet ou non de  $H_0$  selon laquelle les coefficients beta de pente sont égal à 0). Si nous prenons les problèmes médicaux, le coefficient beta de la pente est de .32 (et significativement différent de 0 car  $p < .05$ ). Cela signifie que, lorsque les problèmes médicaux augmentent d'une unité, la consommation de drogue augmente de .32 unités.

En bleu, les **statistiques générales de notre modèle** (pourcentages de variance expliquée et significativité). Pour rappel, ce modèle considère précisément les quatre problèmes comme prédicteurs et la consommation de drogue comme critère. Le **multiple  $R^2$**  est la variance du critère expliquée par les prédicteurs dans l'échantillon, et le  **$R^2$  ajusté** ce même pourcentage de variance estimé pour la population.

Dans les tableaux ci-dessus, aucune information n'a encore été donnée sur les **coefficients standardisés  $\beta$**  (betas) de chaque prédicteur, nécessaires pour la notation finale.

**lm.beta(RegMult)**

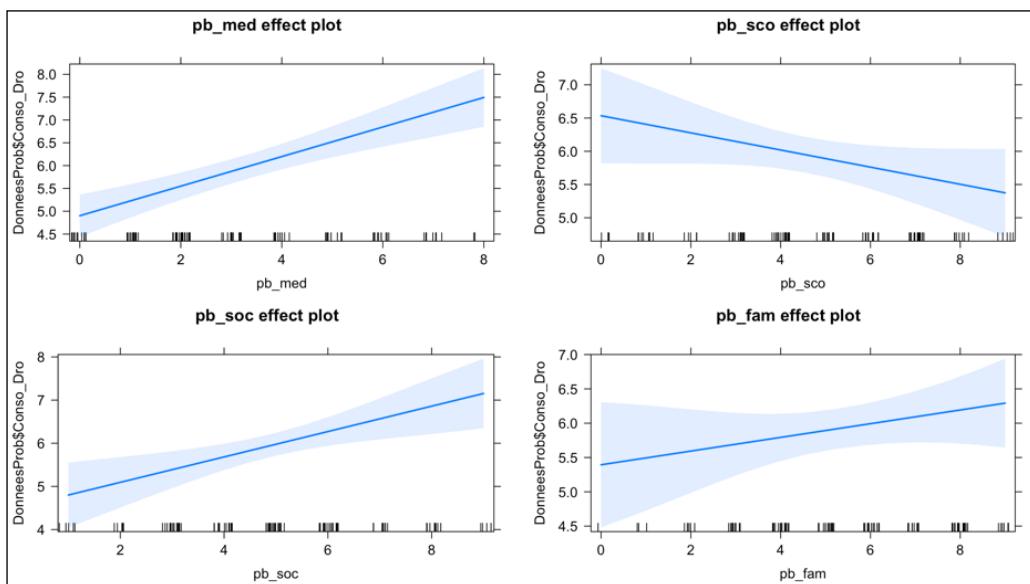
```
Call:
lm(formula = Conso_Dro ~ pb_med + pb_sco + pb_soc + pb_fam, data = Database)

Standardized Coefficients:
(Intercept)      pb_med      pb_sco      pb_soc      pb_fam
0.00000000  0.4344981 -0.1829438  0.3692237  0.1367502
```

Nous voyons que ces coefficients standardisés sont différents des coefficients de la pente. En effet, ils sont standardisés de façon à pouvoir être comparés aux autres prédicteurs lorsque ces derniers ne partagent pas la même échelle. Cette indice  $\beta$  représente une corrélation entre le prédicteur en question et le critère (donc ses valeurs se situent donc entre  $-1$  et  $+1$ ), et sa valeur est proche de la corrélation semi-partielle, parce qu'il représente l'apport spécifique du prédicteur au critère (l'effet des autres prédicteurs étant contrôlé statistiquement).

Pour mieux interpréter les résultats, il est possible d'observer les **graphiques des effets séparés** des prédicteurs avec le critère.

**plot(allEffects(RegMult))**



Ces graphiques peuvent faciliter l'interprétation des données. Concernant les problèmes médicaux, il est possible de voir que plus les participant·es ont de problèmes médicaux, plus leur consommation de drogue est élevée. La zone bleutée entourant la droite de prédiction représente l'intervalle de confiance (si nous prenons les problèmes familiaux, la variabilité par rapport à la droite de prédiction est très importante, ce qui risque de diminuer sa significativité au sein du modèle).

Finalement, l'**équation de la droite de régression**, permet de résumer notre modèle sous forme de chiffres. Ces valeurs se trouvent également dans le tableau de régression obtenu précédemment sous la colonne « Estimate ».

**coef(RegMult)**

(Intercept)	pb_med	pb_sco	pb_soc	pb_fam
3.54821051	0.32400266	-0.12888314	0.29421560	0.09979606

Ces coefficients permettent d'écrire l'équation de la droite de régression (sous forme  $y = a \cdot x + b$ ). Il est seulement nécessaire d'indiquer les coefficients pour lesquels le prédicteur était significatif dans le modèle et le coefficient de l'abscisse à l'ordonnée si ce dernier était significatif également.

$$\text{Consommation de drogue} = 3.55 + .32 \cdot pb\_med + .29 \cdot pb\_soc$$

**Notation :** « Une analyse de régression a été menée afin de savoir quels types de problèmes (familiaux, scolaires, médicaux ou sociaux) prédisaient la gravité de la consommation de drogue chez des adolescent·es. Les prérequis de l'analyse ont été satisfaits. Les quatre prédicteurs du modèle expliquent 37% de la variance du critère dans la population,  $R^2_{adj} = .37$ , ce qui est significativement différent de 0,  $F(4, 94) = 15.12, p < .05$ . L'importance des problèmes médicaux,  $\beta = .43, t = 5.31, p < .05$ , et le nombre de problèmes sociaux,  $\beta = .37, t = 3.19, p < .05$ , prédisent significativement et positivement la gravité de la consommation de drogue. Plus les problèmes médicaux et plus le nombre de problèmes familiaux sont importants, plus la consommation de drogue sera élevée. Cependant, les problèmes scolaires,  $\beta = -.18, t = -1.81, p > .05$ , et les problèmes familiaux,  $\beta = .14, t = 1.22, p > .05$ , ne prédisent pas significativement la consommation de drogue dans le modèle. »

#### 4.4.8.2 Régression linéaire incluant des variables nominales

Packages supplémentaires nécessaires : {ggplot2}

La procédure de régression avec une variable nominale est presque identique à celle avec uniquement des variables quantitatives. Cependant, il est nécessaire de prêter attention à l'interprétation de la valeur nominale, car l'interprétation dépend de l'ordre dans lequel ont été codées les modalités.

Exemple : Nous souhaitons savoir si le sexe (`Sexe`) et les problèmes sociaux (`pb_soc`) prédisent les problèmes familiaux (`pb_fam`) rencontrés par des adolescent·es, variables tirées de la base de données `Database`. L'analyse à effectuer est une **régression linéaire multiple** avec **deux prédicteurs**, dont un nominal (à deux modalités) et l'autre quantitatif, et la consommation de drogue comme **critère**.

```
RegMult_nom <- lm(pb_fam ~ Sexe + pb_soc, data=DonneesProb)
summary(RegMult_nom) ; lm.beta(RegMult_nom)
```

```
Call:
lm(formula = pb_fam ~ sexe + pb_soc, data = Database)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.9667 -1.0694 -0.1874  1.1936  3.9306 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.59524   0.41478   3.846 0.000216 ***
sexeFemme   1.11803   0.33540   3.333 0.001220 **  
pb_soc       0.69483   0.07895   8.801 5.6e-14 *** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.587 on 96 degrees of freedom
Multiple R-squared:  0.5097,    Adjusted R-squared:  0.4995 
F-statistic: 49.9 on 2 and 96 DF,  p-value: 1.384e-15
```

Dans les tableaux ci-dessus, aucune information n'a encore été donnée sur les **coefficients standardisés  $\beta$**  de chaque prédicteur, nécessaires pour la notation finale.

```
lm.beta(RegMult_nom)
```

```
Call:
lm(formula = pb_fam ~ sexe + pb_soc, data = Database)

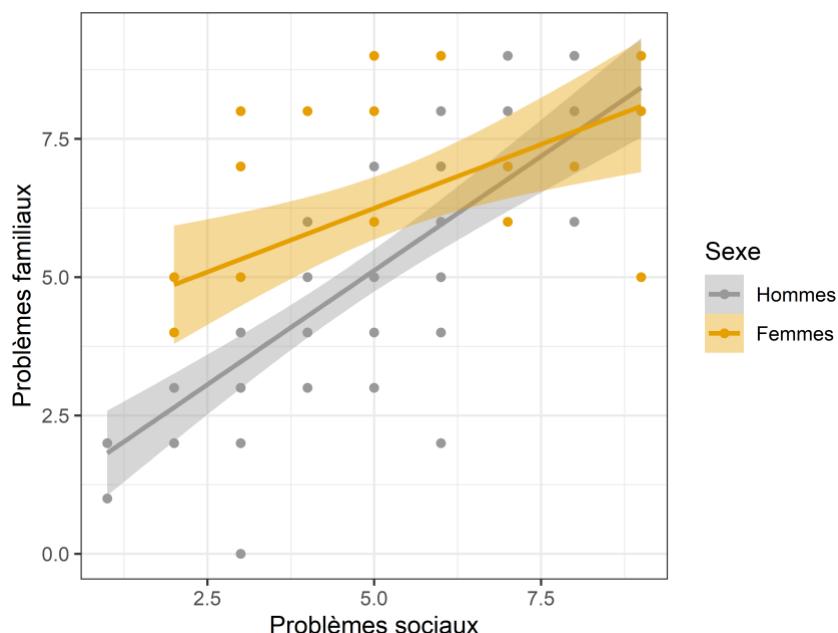
Standardized Coefficients:
(Intercept) sexeFemme pb_soc
0.0000000  0.2410314  0.6363346
```

Pour interpréter correctement le modèle de régression, il est fondamental de connaître la **modalité de référence du facteur dans le modèle de régression**. En l'occurrence, et comme les modalités du facteur ont été renommées, la variable « Sexe » est accolée à la modalité « Femme » (codé 1), ce qui signifie que la référence est la modalité « Homme » (codé 0). Si le codage est inverse, le beta sera pareil mais de signe inversé.

Une manière d'illustrer ce phénomène est de voir les coefficients non standardisés comme un ajout ou une diminution de la valeur moyenne du critère pour la modalité présentée. En l'occurrence, nous pouvons constater que les femmes possèdent en moyenne 1.12 unités supplémentaires de problèmes familiaux en plus que les hommes.

**Illustrer graphiquement un modèle de régression comprenant une variable nominale :** La commande suivante permet d'observer l'écart entre hommes et femmes sur le lien entre problèmes sociaux et problèmes familiaux.

```
ggplot(Database, aes(x=pb_soc, y=pb_fam, fill=Sexe, color=Sexe)) +
  geom_point(size = 1.5) +
  xlab("Problèmes sociaux") + ylab("Problèmes familiaux") +
  scale_color_manual(values=c('#999999', '#E69F00')) +
  scale_fill_manual(values = c('#999999', '#E69F00')) +
  geom_smooth(method=lm) +
  theme_bw()
```



**Notation :** « Une analyse de régression a été menée afin de savoir si les problèmes sociaux et le sexe prédisent l’importance des problèmes familiaux chez des adolescent·es. Les prérequis de l’analyse ont été satisfaits. Les prédicteurs du modèle expliquent 49% de la variance du critère dans la population,  $R^2$  ajusté = .49, ce qui est significativement différent de 0,  $F(2, 96) = 49.90, p < .05$ . L’importance des problèmes sociaux,  $\beta = .64, t = 8.80, p < .05$ , et le sexe des adolescent·es,  $\beta = .24, t = 3.33, p < .05$ , prédisent significativement et positivement la gravité de la consommation de drogue. Les filles montrent davantage de problèmes familiaux que les garçons, et plus le nombre de problèmes sociaux est élevé, plus la consommation de drogue l'est aussi. »

#### 4.4.8.3 Méthode de sélection Stepwise (Forward, Backward)

Package et fonction supplémentaire nécessaire : ols\_step\_both\_p {olsrr}

Dans la régression linéaire, la méthode de sélection *Stepwise* (ou *pas-à-pas*) consiste à ajouter (*Forward*) et, si nécessaire, retirer (*Backward*) des variables sur différents modèles afin de les comparer et d’en obtenir le meilleur. Le meilleur de ces modèles consistera ainsi en l’ensemble le plus représentatif des variables (généralement significatives) pour la prédiction du critère tout en diminuant l’erreur de prédiction (retirant les variables qui ne sont pas significatives).

La **méthode de sélection Stepwise ou remplacement séquentiel** consiste en la combinaison des deux méthodes *Forward* et *Backward*. Le premier modèle sélectionné sera un modèle nul, soit le critère seul sans prédicteur. Puis, séquentiellement, les variables expliquant le plus de variance seront ajoutées dans le modèle (*Forward*). Finalement, après l’ajout de chaque variable, seront retirées du modèle (*Backward*) celles qui ne seront plus significatives (généralement dû à une multicolinéarité entre variables).

Exemple : Nous souhaitons savoir quels types de problèmes (médicaux sous `pb_med`, scolaires sous `pb_sco`, sociaux sous `pb_soc` et/ou familiaux sous `pb_fam`) prédisent la consommation de drogue (`Conso_Dro`) chez des adolescent·es, variables tirées de la base de données `Database`. Nous souhaitons obtenir le meilleur modèle de régression sans variables non pertinentes (non significatives). L’analyse à effectuer est une **régression linéaire multiple avec méthode de sélection Stepwise**.

```
RegMult <- lm(Conso_Dro ~ pb_med + pb_sco + pb_soc + pb_fam,
               data=Database) #Modèle sans Stepwise
RegMultSTEP <- ols_step_both_p(RegMult, details=TRUE) #Stepwise
```

```

Stepwise Selection Method
-----
Candidate Terms:
1. pb_med
2. pb_sco
3. pb_soc
4. pb_fam

We are selecting variables based on p value...

Stepwise Selection: Step 1

✓ pb_med

      Model Summary
-----
R          0.492    RMSE       1.432
R-Squared  0.242    Coef. Var  24.154
Adj. R-Squared 0.234    MSE        2.051
Pred R-Squared 0.212    MAE        1.180
-----
RMSE: Root Mean Square Error
MSE: Mean Square Error
MAE: Mean Absolute Error

      ANOVA
-----
      Sum of
      Squares   DF   Mean Square   F     Sig.
-----
Regression  63.546   1    63.546   30.981  0.0000
Residual    198.959  97   2.051
Total       262.505  98
-----
      Parameter Estimates
-----
      model   Beta   Std. Error   Std. Beta   t     Sig   lower   upper
-----
(Intercept) 4.766   0.254
pb_med      0.367   0.066   0.492    5.566  0.000   0.236   0.498
-----
```

Descriptif de la méthode de sélection et variables

**Etape 1<sup>ère</sup>:**  
Ajout de la variable pb\_med dans le modèle

```

Stepwise Selection: Step 2

✓ pb_soc

      Model Summary
-----
R          0.605    RMSE       1.316
R-Squared  0.366    Coef. Var  22.201
Adj. R-Squared 0.353    MSE        1.733
Pred R-Squared 0.328    MAE        1.078
-----
RMSE: Root Mean Square Error
MSE: Mean Square Error
MAE: Mean Absolute Error

      ANOVA
-----
      Sum of
      Squares   DF   Mean Square   F     Sig.
-----
Regression  96.161   2    48.081   27.748  0.0000
Residual    166.344  96   1.733
Total       262.505  98
-----
      Parameter Estimates
-----
      model   Beta   Std. Error   Std. Beta   t     Sig   lower   upper
-----
(Intercept) 3.530   0.368
pb_med      0.322   0.061   0.431    5.234  0.000   0.200   0.444
pb_soc      0.285   0.066   0.358    4.339  0.000   0.155   0.415
-----
```

**Etape 2<sup>ème</sup>:**  
Ajout de la variable pb\_soc dans le modèle

No more variables to be added/removed.

Final Model Output

Model Summary					
R	0.605	RMSE	1.316		
R-Squared	0.366	Coef. Var	22.201		
Adj. R-Squared	0.353	MSE	1.733		
Pred R-Squared	0.328	MAE	1.078		

RMSE: Root Mean Square Error
MSE: Mean Square Error
MAE: Mean Absolute Error

ANOVA					
	Sum of Squares	DF	Mean Square	F	Sig.
Regression	96.161	2	48.081	27.748	0.0000
Residual	166.344	96	1.733		
Total	262.505	98			

Parameter Estimates							
model	Beta	Std. Error	Std. Beta	t	Sig	lower	upper
(Intercept)	3.530	0.368		9.587	0.000	2.799	4.261
pb_med	0.322	0.061	0.431	5.234	0.000	0.200	0.444
pb_soc	0.285	0.066	0.358	4.339	0.000	0.155	0.415

**Modèle final :**  
Aucune autre prédicteur n'est à insérer.  
**Modèle à prendre en considération.**

Dans cet exemple, seules deux étapes ont été nécessaires pour obtenir le meilleur modèle de régression. A chaque étape, toutes les informations nécessaires pour décrire le modèle en cours sont décrites (résumé du modèle, test statistique du modèle, tests statistiques des prédicteurs y étant inclus).

**Notation :** « Une analyse de régression avec méthode de sélection Stepwise a été menée afin de savoir quels types de problèmes (familiaux, scolaires, médicaux ou sociaux) prédisent la gravité de la consommation de drogue chez des adolescent·es. Les prérequis de l'analyse ont été satisfaits. Deux prédicteurs ont été retenus dans le modèle final par la méthode de sélection et expliquent 35% de la variance du critère dans la population,  $R^2$  ajusté = .35, ce qui est significativement différent de 0,  $F(2, 96) = 27.75, p < .05$ . Seules l'importance des problèmes médicaux,  $\beta = .43, t = 5.23, p < .05$ , et des problèmes sociaux,  $\beta = .36, t = 4.34, p < .05$ , prédisent significativement et positivement la gravité de la consommation de drogue. Plus les problèmes médicaux et plus le nombre de problèmes familiaux sont importants, plus la consommation de drogue sera élevée. »

#### 4.4.8.4 Méthode de sélection hiérarchique

Package et fonction supplémentaire nécessaire : anova {stats}

Remarque : Une commande pour un tableau de résultats plus complet (mais sans valeur de F) est disponible au chapitre 5.4 Régression linéaire (Situation 3).

A la différence de la méthode de sélection *Stepwise* qui choisit les critères sur des bases mathématiques, la méthode de sélection hiérarchique permet à la chercheuse ou au chercheur de déterminer l'ordre d'entrée des variables dans des divers blocs (ou modèles). Ces blocs sont ensuite comparés hiérarchiquement en fonction de leur entrée dans le modèle selon s'ils expliquent plus ou moins de variance du critère. Cette méthode permet de tester des modèles théoriques connus ou de sélectionner des blocs de prédicteurs de nature identique (p. ex.: prédicteurs socio-démographiques vs. prédicteurs attitudinaux). Les recherches davantage exploratoires privilégieront la méthode de sélection *Stepwise*.

Exemple : Nous souhaitons savoir si un premier bloc de variables socio-démographiques incluant l'âge (`Age`), le genre (`Gender`) et l'orientation politique (`OrienPol`) prédisent les explications complotistes de l'économie (p. ex.: « Les crises économiques ont été délibérément créées par certain·es pour s'enrichir. », `ExplC`). Dans un second bloc, nous souhaitons y entrer des variables d'attitude incluant la satisfaction dans la vie (`Sat`), la méfiance politique (`DtrustPol`), le sentiment de manque de contrôle (`LackCtrl`), l'autoritarisme de droite (`RWA`), la croyance en un moment dangereux (`BDW`) et l'adhésion aux théories du complot (`TC`), variables tirées de la base de données `Database`. Par ces deux blocs, nous souhaitons savoir si un second bloc incluant des mesures d'attitude prédit significativement plus de variance des explications complotistes de l'économie qu'un bloc n'incluant que des mesures socio-démographiques. Nous avons donc **deux blocs** insérés **hiérarchiquement** dans un modèle de régression linéaire.

**Entrée des blocs :** Les commandes restent identiques à la régression multiple développée précédemment. Cependant, nous allons créer deux modèles de régression au lieu d'un. Il est nécessaire d'insérer un seul et même critère dans chacun des blocs. Dans notre exemple, nous créons seulement deux blocs, mais il est tout à fait possible d'en créer davantage.

```
## Bloc 1 : Mesures sociodemographiques
RM_Mod01 <- lm(ExplC ~ Age + Gender + OrienPol,
                 data = Database)
## Bloc 2 : Mesures sociodemographiques + Mesures d'attitude
RM_Mod02 <- lm(ExplC ~ Age + Gender + OrienPol +
                 Sat + DtrustPol + LackCtrl + RWA + BDW + TC,
                 data = Database)
```

**Analyse hiérarchique des blocs :** Une fois les blocs créés, la prochaine étape consiste à les comparer. Veillez à insérer les blocs de manière que chaque bloc soit comparé au précédent qui lui correspond : soit une composition telle que `anova(bloc01, bloc02, bloc03)` et non de manière désordonnée.

```
anova(RM_Mod01, RM_Mod02)
```

```

Analysis of Variance Table

Model 1: ExplC ~ Age + Gender + OrienPol
Model 2: ExplC ~ Age + Gender + OrienPol + Satisfaction + DistrustPol +
          LackControl + RWA + BDW + TC
Res.Df   RSS Df Sum of Sq    F    Pr(>F)
1      56 23.387
2      50 11.765  6   11.622 8.2323 3.25e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Les résultats nous indiquent que le second modèle, incluant à la fois les mesures socio-démographiques *et* les mesures d'attitude, expliquent une part significativement plus importante qu'un modèle n'incluant que des mesures socio-démographiques.

**Important :** Dans l'analyse de régression hiérarchique, nous ne nous intéressons pas seulement à savoir si l'un ou l'autre modèle prédit significativement ou non le critère, comme précédemment. Nous cherchons également à savoir si le *change* ou passage d'un modèle à un autre permet d'expliquer une part plus importante de la variance du critère que le modèle précédent. Pour ce faire, nous allons comparer les  $R^2$ , et soustraire le  $R^2$  du modèle 1 à celui du modèle 2, afin de calculer la différence de  $R^2$  gagnée par le modèle 2 sur le modèle 1 (ce qui se note  $\Delta R^2$  et qui signifie différence de  $R^2$ ). Afin de savoir si ce  $\Delta R^2$  est significativement différent de 0, nous utilisons à nouveau un test  $F$  nommé  $F_{change}$  (ou  $\Delta F$ ). Pour avoir les  $R^2$  de chaque modèle, il est nécessaire de les calculer séparément comme sur la commande ci-dessous.

```

summary(RM_Mod01) #Modele 1 a R2 = .05
summary(RM_Mod02) #Modele 2 a R2 = .52 -> DR2 = .47

```

```

Residual standard error: 0.6462 on 56 degrees of freedom
(229 observations deleted due to missingness)
Multiple R-squared:  0.04597,  Adjusted R-squared:  -0.00514
F-statistic: 0.8994 on 3 and 56 DF,  p-value: 0.4473

```

```

Residual standard error: 0.4851 on 50 degrees of freedom
(229 observations deleted due to missingness)
Multiple R-squared:  0.5201,  Adjusted R-squared:  0.4337
F-statistic:  6.02 on 9 and 50 DF,  p-value: 1.117e-05

```

**Notation :** « Une analyse de régression avec méthode de sélection hiérarchique a été menée en incluant dans un premier bloc des mesures socio-démographiques (âge, genre et orientation politique), qui n'explique pas de part de variance significative des explications complotistes de l'économie,  $R^2 = .05$ ,  $F(3, 56) = .90$ ,  $p > .05$ . Cependant, en ajoutant les mesures d'attitude (satisfaction dans la vie, méfiance politique, manque de contrôle, autoritarisme de droite, croyance en un monde dangereux et adhésion aux théories du complot), le modèle explique une part significative de variance du critère,  $R^2 = .52$ ,  $F(9, 50) = 6.02$ ,  $p < .05$ . Le passage du modèle 1 au modèle 2 est significatif,  $\Delta R^2 = .47$ ,  $\Delta F(6, 50) = 8.23$ ,  $p < .05$ . »

#### 4.4.8.5 Analyse des résidus

Fonction et packages nécessaires : `shapiro.test {stat}` ; `ncvTest {car}`

Outre la normalité uni- et bivariée, le nombre de sujets et la présence de corrélations entre critère et prédicteurs, l'analyse des résidus<sup>1</sup> est également une condition d'application à prendre en considération dans l'analyse de régression linéaire. Cette analyse permet de tester la validité d'un modèle. En effet, même si une valeur  $p$  est très proche de 0, cette valeur ne permet pas de savoir si le modèle comporte quelque anomalie. Pour cela, la méthode graphique est la plus adaptée. Prenons l'analyse de régression multiple développée aux chapitres précédents.

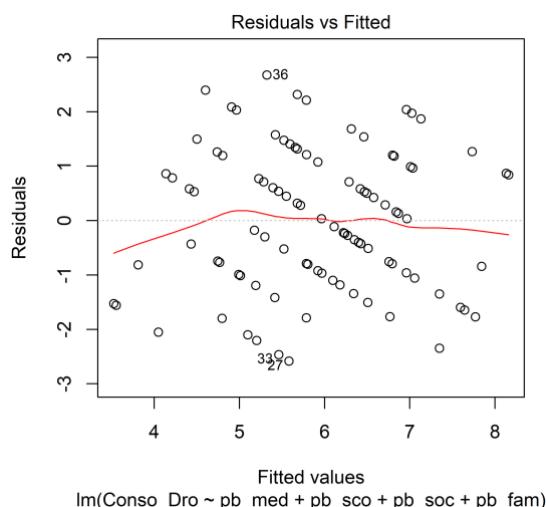
#### Condition 1 : Linéarité de la régression (valeurs prédictées vs. résidus)

Ce premier graphique permet de savoir si les valeurs prédictées (ou *fitted values*, en x) et les résidus (ou *residuals*, en y) suivent une distribution bivariée linéaire. Pour que la condition de linéarité du modèle soit respectée, le segment représenté en rouge doit se situer, idéalement, sur une valeur résiduelle (ou y) égale à 0. Nous pouvons considérer cette condition respectée.

```
plot(RegMult, which=1) # 1 = Residuals vs Fitted
```

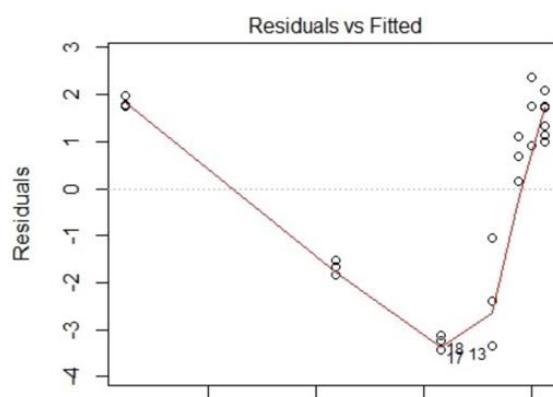
##### Exemple de linéarité

Ci-dessous, un exemple où l'hypothèse de la linéarité de la régression est respectée.



##### Exemple de non linéarité

Ci-dessous, un exemple où l'hypothèse de la linéarité de la régression serait largement rejetée.



#### Condition 2 : Normalité des résidus

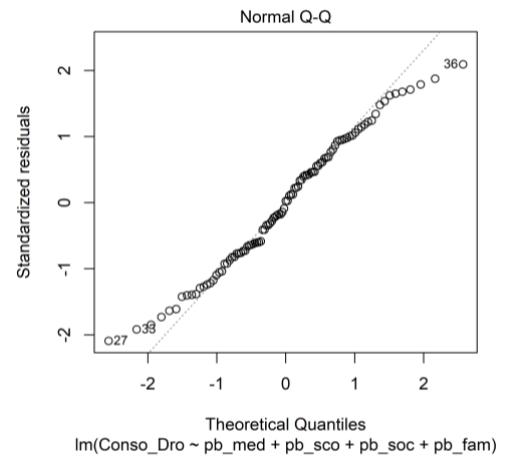
Ce deuxième graphique permet de savoir si les résidus sont normalement distribués (en fonction de quantiles théoriques, d'où le « Q-Q »). Pour que la condition de normalité des résidus soit respectée, les résidus doivent suivre la droite pointillée. Nous pouvons considérer cette condition respectée.

```
plot(RegMult, which=2) # 2 = Normal Q-Q
```

<sup>1</sup> Un **résidu** est la différence entre une valeur observée et une valeur prédictée par l'équation de régression. Les résidus représentent la variation inexplicable par le modèle.

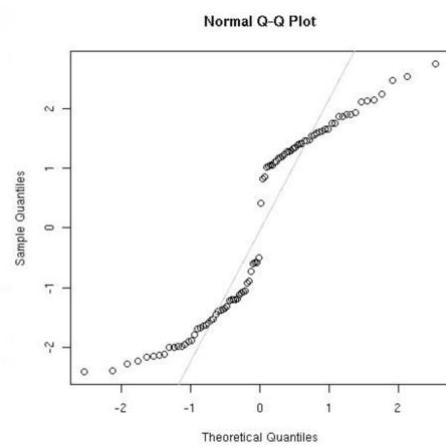
### Exemple de normalité des résidus

Ci-dessous, un exemple où l'hypothèse de normalité des résidus est respectée.



### Exemple de non normalité des résidus

Ci-dessous, un exemple où l'hypothèse de normalité des résidus n'est pas respectée.



L'**analyse du test de Shapiro-Wilk** est une alternative pour évaluer la normalité des résidus. Ce test doit être non significatif pour considérer la normalité des résidus comme respectée. Dans notre exemple : selon le test de Shapiro-Wilk, et conjointement au graphique précédent, nous pouvons considérer la normalité des résidus comme respectée,  $W = .98, p > .05$ .

```
shapiro.test(residuals(RegMult))
```

```
Shapiro-Wilk normality test  
data: residuals(RegMult)  
W = 0.98112, p-value = 0.167
```

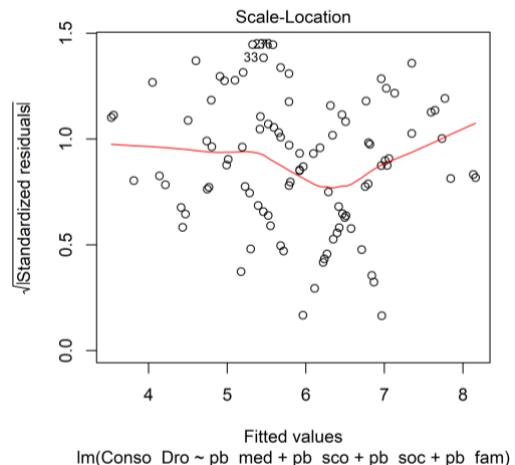
### Condition 3 : Homoscédasticité ou Homogénéité des résidus

Ce troisième graphique permet de savoir si les résidus ont une variance égale tout au long des valeurs prédictes par la droite de régression. Il permet ainsi de vérifier si la condition d'application de l'homogénéité de la variance est respectée (homoscédasticité) ou non (hétéroscédasticité). Pour que la condition d'homogénéité des résidus soit respectée, le segment représenté en rouge doit être, idéalement, le plus droit et horizontal possible. Dans notre exemple, l'homogénéité semble être relativement respectée bien qu'elle montre une certaine déformation à partir de la valeur prédictive « 6 » (en x). En cas de doute, il est utile de se référer au test de Breush-Pagan.

```
plot(RegMult, which=3) # 3 = Scale Location
```

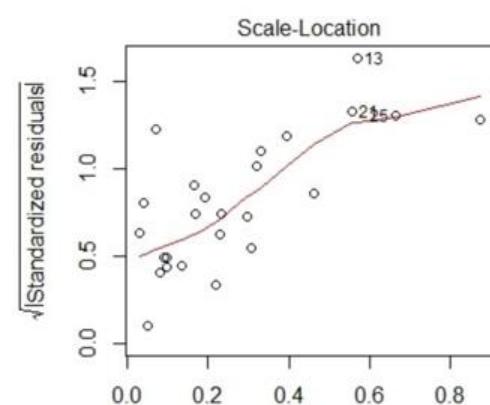
### Exemple d'homoscédasticité

Ci-dessous, un exemple où l'hypothèse d'homoscédasticité relativement respectée.



### Exemple d'hétéroscedasticité

Ci-dessous, un exemple où l'hypothèse d'homoscédasticité n'est pas respectée.



L'**analyse du test de Breush-Pagan** est une alternative pour évaluer l'homogénéité des résidus. Ce test doit être non significatif pour considérer la normalité des résidus comme respectée. Dans notre exemple : selon le test de Breush-Pagan, et conjointement au graphique précédent (à gauche), nous pouvons considérer l'homogénéité des résidus comme respectée,  $\chi^2(1) = .65, p > .05$ .

```
ncvTest(RegMult)
```

```
Non-constant Variance Score Test  
Variance formula: ~ fitted.values  
Chisquare = 0.6512333, Df = 1, p = 0.41967
```

### Condition 4 : Indépendance des résidus

L'indépendance des résidus est à tester lorsque les données proviennent d'une étude longitudinale (avec la récolte de données répétée d'un même individu sur une certaine période de temps). Ces études s'opposent, conceptuellement, aux études transversales dont la récolte se fait une seule fois et dont la condition d'indépendance des résidus est considérée, par défaut, comme respectée.

#### 4.4.8.6 Modération

Une variable modératrice (ou modérateur) est une variable qui altère la force, et possiblement la direction, du lien entre un prédicteur et un critère. Ainsi, l'effet du prédicteur sur le critère dépend également des niveaux de cette variable modératrice. La modération est l'équivalent de l'effet d'interaction de l'ANOVA pour la régression. Comme exemple, nous pouvons supposer qu'un lien direct entre le genre et les performances mathématiques soit en réalité *modéré* par la menace du stéréotype. La variable modératrice peut être une variable nominale (comme le genre), mais également une variable continue (comme le niveau de neuroticisme). Ces deux possibilités sont développées dans les sous-chapitres suivants.

##### 4.4.8.6.1 Modération par une variable nominale

Fonctions et packages supplémentaires nécessaires : plotSlopes {rockchalk}

Exemple : Nous souhaitons savoir si l'effet des problèmes sociaux (`pb_soc`) sur les problèmes familiaux (`pb_fam`) est modéré par le sexe (`Sexe`) des adolescent·es, variables tirées de la base de données `Database`. Nous avons donc **un prédicteur**, **un critère** et **une variable modératrice nominale**. Notez que cet exemple est presque identique à celui développé au chapitre *Régression linéaire incluant des variables nominales* (4.4.8.2), à l'exception que nous ne considérons le sexe plus seulement comme un prédicteur, mais également comme un modérateur. L'interaction est spécifiée en inscrivant une \* entre prédicteur et modérateur à la place du +.

```
RegModerN <- lm(pb_fam ~ pb_soc*Sexe, data=Database)
summary(RegModerN)
```

```
Call:
lm(formula = pb_fam ~ pb_soc * Sexe, data = Database)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.9466 -0.9466 -0.1212  1.3240  3.8788 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.99424   0.48555  2.048  0.04335 *  
pb_soc       0.82539   0.09653  8.551 2.06e-13 *** 
SexeFemme    2.94589   0.87321  3.374  0.00108 ** 
pb_soc:SexeFemme -0.36425   0.16123 -2.259  0.02616 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 1.554 on 95 degrees of freedom
Multiple R-squared:  0.5347,    Adjusted R-squared:  0.52 
F-statistic: 36.39 on 3 and 95 DF,  p-value: 9.516e-16
```

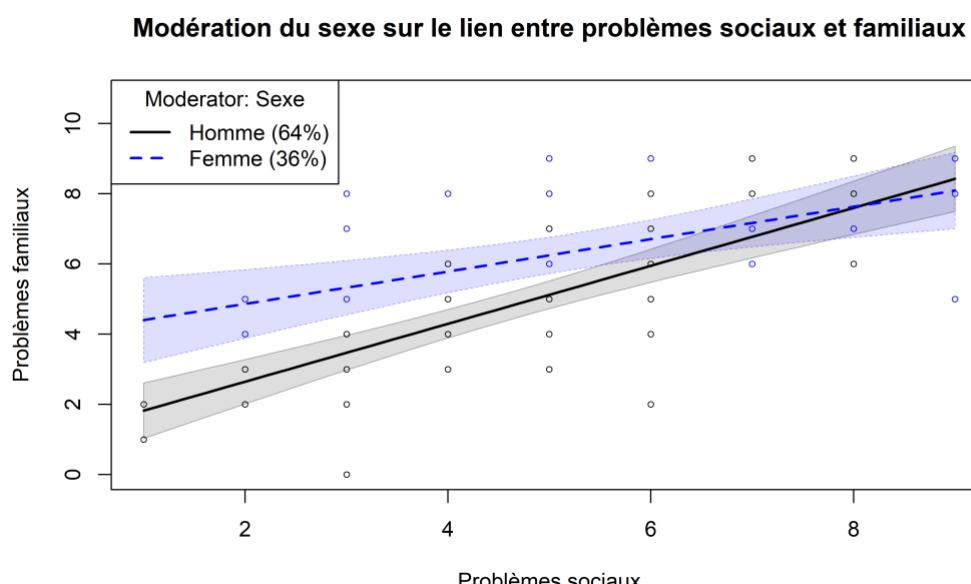
```
Standardized Coefficients:
                (Intercept)          pb_soc        SexeFemme pb_soc:SexeFemme
                0.00000000       0.7559063      0.6350917      -0.4570849
```

Nous constatons que l'interaction (en rouge) est significative dans le modèle de régression, soit  $t = -2.26$ ,  $p < .05$ . Ainsi, le lien ou la force du lien entre le prédicteur (les problèmes sociaux) et le critère (les problèmes familiaux) est différente selon le sexe des adolescent·es.

L’illustration graphique de cette modération peut être faite très simplement grâce à la fonction `plotSlopes`. Cette fonction demande que l’analyse de régression soit inscrite de la même manière que ci-dessus, soit `critère ~ prédicteur * modérateur`. Voici les différents éléments à insérer :

- Sous `plotx=`, le prédicteur doit être spécifié.
- Sous `modx=`, la variable modératrice doit être spécifiée.
- Par `interval="confidence"`, la variabilité des droites de régression sera donnée par l’intervalle de confiance à 95%.

```
plotSlopes(RegModerN, plotx="pb_soc", modx="Sexe",
            interval="confidence", main="Graphique de moderation",
            xlab="Problemes sociaux", ylab="Problemes familiaux")
```



Le graphique ci-dessus nous permet de visualiser la modération du sexe sur l’effet des problèmes sociaux sur les problèmes familiaux. L’effet des problèmes sociaux demeure plus marqué chez les garçons que chez les filles.

**Notation :** « Une analyse de modération a été effectuée afin de savoir si la prédiction des problèmes familiaux par les problèmes sociaux est modérée par le sexe des adolescent·es. Les problèmes sociaux prédisent significativement et positivement les problèmes familiaux,  $\beta = .76$ ,  $t = 8.55$ ,  $p < .05$ . Le sexe (modalité de référence : homme) prédit significativement les problèmes familiaux,  $\beta = .64$ ,  $t = 3.37$ ,  $p < .05$ . L’effet d’interaction entre les problèmes sociaux et le sexe prédit significativement les problèmes familiaux,  $\beta = -.46$ ,  $t = -2.26$ ,  $p < .05$ . Une modération du sexe est remarquable graphiquement : la prédiction positive des problèmes sociaux est davantage marquée chez les adolescents, moins chez les adolescentes, bien que ces dernières reportent en moyenne davantage de problèmes familiaux. »

#### 4.4.8.6.2 Modération par une variable continue

Fonctions et packages supplémentaires nécessaires : plotSlopes {rockchalk}

Exemple : Nous souhaitons savoir si l'effet du neuroticisme (`epiNeur`) sur la dépression (`bdi`) est modéré par les traits anxieux (`stateanx`), variables tirées de la base de données `Database`<sup>1</sup>. Nous avons donc **un prédicteur, un critère et une variable modératrice continue** (ou **quantitative**). Toutes ces mesures ont été évaluées par le moyen d'échelles cliniques standardisées. L'interaction est spécifiée par une \* entre prédicteur et modérateur.

**Remarque :** Il est nécessaire de **centrer les variables quantitatives prédictrices** (prédicteur et modérateur), soit soustraire chacune de ces mesures par leurs moyennes (comme exemplifié pour `epiNeur.ctr`), puis d'en effectuer l'analyse de modération.

```
Database$epiNeur.ctr <- Database$epiNeur - mean(Database$epiNeur,
na.rm=TRUE) #Exemple de Centrage de "epiNeur"
RegModerC <- lm(bdi ~ epiNeur.ctr*stateanx.ctr, data=Database)
summary(RegModerC)
```

```
Call:
lm(formula = bdi ~ epiNeur.ctr * stateanx.ctr, data = Database)

Residuals:
    Min      1Q  Median      3Q     Max 
-12.0493 -2.2513 -0.4707  2.1135 11.9949 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 6.35996   0.29969 21.222 < 2e-16 ***
epiNeur.ctr 0.46123   0.06409  7.197 8.93e-12 ***
stateanx.ctr 0.19658   0.02772  7.091 1.67e-11 ***
epiNeur.ctr:stateanx.ctr 0.01528   0.00466  3.279  0.0012 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.12 on 227 degrees of freedom
Multiple R-squared:  0.4978, Adjusted R-squared:  0.4912 
F-statistic: 75.02 on 3 and 227 DF,  p-value: < 2.2e-16
```

Standardized Coefficients::	(Intercept)	epiNeur.ctr	stateanx.ctr	epiNeur.ctr:stateanx.ctr
0.0000000	0.3913033	0.3908944	0.1576503	

Nous constatons que l'interaction (en rouge) est significative dans le modèle de régression, soit  $t = 3.28$ ,  $p < .01$ . Ainsi, la force du lien entre le prédicteur (neuroticisme) et le critère (dépression) est altérée par les traits anxieux.

**L'illustration graphique de cette modération** peut être faite très simplement grâce à la fonction `plotSlopes`. Cette fonction demande que l'analyse de régression soit inscrite de la même manière que ci-dessus, soit `critère ~ prédicteur * modérateur` (sans centrage). Voici les différents éléments à insérer :

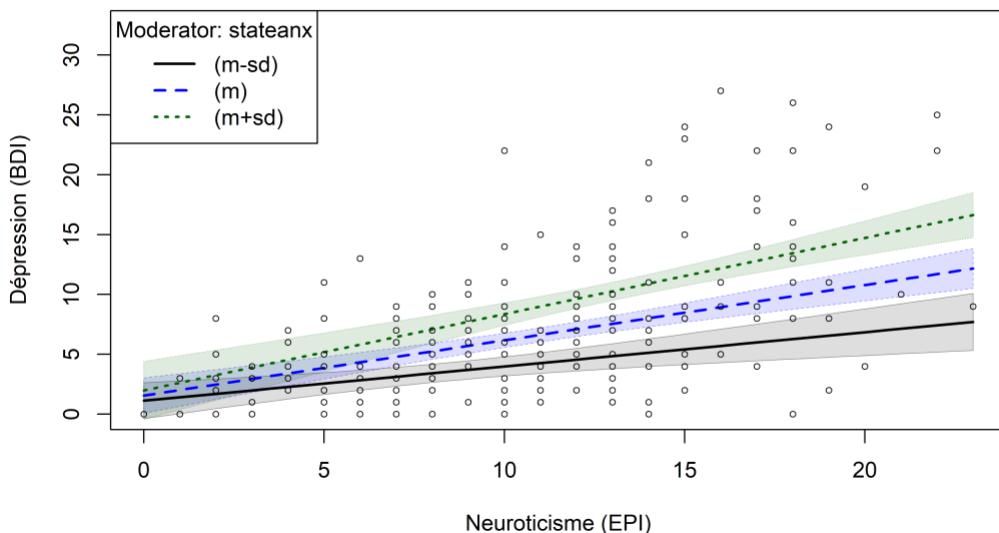
- Sous `plotx=`, le prédicteur doit être spécifié.
- Sous `modx=`, la variable modératrice doit être spécifiée.
- Par `interval="confidence"`, la variance des droites de régression sera donnée par l'intervalle de confiance à 95%.

<sup>1</sup> Cette base de données est disponible sur le logiciel R grâce au package `{psych}`. Une fois le package activé, vous pouvez importer cette base de données en compilant `data(epi.bfi)`. La base de données a été ensuite renommée `Database` pour des commodités de lecture des commandes.

- Par `modxVals="std.dev"`, la modération par la variable quantitative sera représentée sous trois niveaux : traits d'anxiété élevés ( $m + sd$ ), modérés ( $m$ ), et bas ( $m - sd$ ).

```
plotSlopes(RegModerC, plotx="epiNeur", modx="stateanx",
           modxVals="std.dev", interval="confidence",
           main="Graphique de moderation",
           xlab="Neuroticisme (EPI)", ylab="Dépression (BDI)")
```

### Modération des traits anxieux sur le lien entre neuroticisme et dépression



Le graphique ci-dessus nous permet de visualiser la modération des traits anxieux sur l'effet du neuroticisme sur la dépression. L'effet du neuroticisme sur le niveau de dépression semble être fortement accentué pour les individus à traits anxieux élevés ( $m + sd$ ), modérément pour les individus à traits anxieux modérés ( $m$ ), et nettement moins marqué pour les individus à traits anxieux bas ( $m - sd$ ).

**Notation :** « Une analyse de modération a été effectuée afin de savoir si la prédiction de la dépression par le neuroticisme est modérée par les traits anxieux. Le neuroticisme,  $\beta = .39$ ,  $t = 7.20$ ,  $p < .05$ , et les traits anxieux,  $\beta = .39$ ,  $t = 7.09$ ,  $p < .05$ , prédisent significativement et positivement la dépression. L'effet d'interaction entre le neuroticisme et les traits anxieux prédit significativement la dépression,  $\beta = .16$ ,  $t = 3.28$ ,  $p < .05$ . L'effet du neuroticisme sur le niveau de dépression semble être fortement accentué pour les individus à traits anxieux élevés, modérément pour les individus à traits anxieux modérés, et nettement moins marqué pour les individus à traits anxieux bas. »

#### 4.4.8.7 Médiation

##### Fonctions et packages supplémentaires nécessaires : mediate {mediation}

Une variable médiatrice (ou médiateur) est une variable, généralement quantitative, rendant indirect le lien entre un prédicteur et un critère dans un modèle de régression simple : l'effet entre prédicteur et critère est alors *médié* ou *médiatisé* par la variable médiatrice. Cette dernière doit remplir deux conditions : elle doit être **endogène**, soit être une variable dépendante présente dans un modèle de causalité (ayant au moins une flèche pointant sur elle, ce qui est le cas lorsque nous procédons à une analyse de médiation) et **apporter une information complémentaire**, soit expliquer *de quelle manière* le prédicteur influence le critère.

Exemple : Nous souhaitons savoir si l'effet du sentiment d'efficacité personnelle de l'enseignant (**SEPE**, prédicteur) sur la cohésion de classe (**Cohesion**, critère) est médiée par l'indiscipline de classe (**Indiscipline**, médiateur), variables tirées de la base de données **Database**. Nous avons donc **un prédicteur, un critère et une variable médiatrice**. Plusieurs étapes sont nécessaires pour évaluer une médiation.

#### Etape 1 : Régression du critère (VD) sur le prédicteur (VI)

Cette étape demande de vérifier la présence d'un lien significatif entre le prédicteur et le critère dans un modèle de régression simple. La présence d'un lien significatif est nécessaire pour continuer l'analyse.

```
LM_PredCrit <- lm(Cohesion ~ SEPE, data=Database)
summary(LM_PredCrit)
```

```
Call:
lm(formula = Cohesion ~ SEPE, data = Database)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.8087 -0.6611  0.1730  0.6955  1.7546 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.2789     0.7479   1.710   0.09125 .  
SEPE        0.3733     0.1350   2.765   0.00711 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 0.9638 on 78 degrees of freedom
Multiple R-squared:  0.08925, Adjusted R-squared:  0.07757 
F-statistic: 7.643 on 1 and 78 DF,  p-value: 0.007108
```

Nous constatons que le sentiment d'efficacité personnelle de l'enseignant (ou SEPE) prédit significativement et positivement la cohésion de classe ( $p < .01$ ).

## **Etape 2 : Modèle de régression du médiateur (VD) sur le prédicteur (VI)**

Cette étape demande de vérifier la présence d'un lien significatif entre le prédicteur et le médiateur dans un modèle de régression simple. La présence d'un lien significatif est nécessaire pour continuer l'analyse. Dans le cas où le lien n'est pas significatif, alors la médiation supposée n'est pas existante.

```
LM_PredMed <- lm(Indiscipline ~ SEPE, data=Database)
summary(LM_PredMed)
```

```
Call:
lm(formula = Indiscipline ~ SEPE, data = Database)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.59353 -0.69827  0.02846  0.58663  1.97789 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.1265    0.6507   6.342 1.37e-08 ***
SEPE        -0.3339    0.1175  -2.842  0.00572 **  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8385 on 78 degrees of freedom
Multiple R-squared:  0.09384, Adjusted R-squared:  0.08223 
F-statistic: 8.078 on 1 and 78 DF,  p-value: 0.005716
```

Nous constatons que le SEPE prédit significativement et négativement l'indiscipline de classe ( $p < .01$ ).

## **Etape 3 : Modèle de régression du critère (VD) sur prédicteur (VI<sub>1</sub>) et médiateur (VI<sub>2</sub>)**

Cette étape demande de vérifier la présence d'un lien significatif entre le médiateur et le critère. Le prédicteur est également inséré dans le modèle afin de vérifier si le lien entre le prédicteur et le critère trouvé à l'étape 1 demeure (pas de médiation) ou disparaît partiellement (médiation partielle) ou complètement (médiation complète) en contrôlant l'effet du médiateur.

```
LM_PredMedCrit <- lm(Cohesion ~ SEPE + Indiscipline, data=Database)
summary(LM_PredMedCrit)
```

```
Call:
lm(formula = Cohesion ~ SEPE + Indiscipline, data = Database)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.2540 -0.5666  0.1083  0.5496  1.8820 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.8518    0.8742   3.262  0.00165 ** 
SEPE        0.2460    0.1347   1.827  0.07162 .  
Indiscipline -0.3812    0.1236  -3.085  0.00283 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9151 on 77 degrees of freedom
Multiple R-squared:  0.1894, Adjusted R-squared:  0.1684 
F-statistic: 8.997 on 2 and 77 DF,  p-value: 0.0003081
```

Nous constatons que l'indiscipline de classe prédit significativement et négativement la cohésion ( $p < .01$ ) alors que le SEPE seulement tendanciellement ( $p < .10$ ). Le SEPE prédisait significativement la cohésion dans le modèle simple à la première étape, mais non plus lorsque le médiateur est inséré : conjointement avec modèle simple à la deuxième étape, cela signifie que le lien est effectivement indirect et emprunte un détour par le médiateur.

Dans le cas où le médiateur prédit significativement le critère tout en neutralisant le lien précédemment significatif avec le prédicteur, alors nous parlerons de **médiation complète**. Dans le cas où le prédicteur demeure également significatif, alors nous parlerons plutôt d'une **médiation partielle** (ou **incomplète**). Dans notre exemple, la médiation peut être considérée comme complète, ou partielle si l'on considère le lien tendanciel du prédicteur avec le critère ( $p = .07$ ).

#### **Etape 4 : Crédation du graphique de médiation**

Cette étape consiste à créer un graphique synthétisant les analyses effectuées aux étapes précédentes. Il existe plusieurs méthodes pour créer ce graphique dont la méthode manuelle. Cette méthode demande à insérer manuellement les noms et les différentes valeurs obtenues sous *Estimate*. Ci-dessous, le graphique illustrant notre exemple. Vous pourrez alors simplement remplacer les noms et les valeurs en fonction de vos données et résultats.

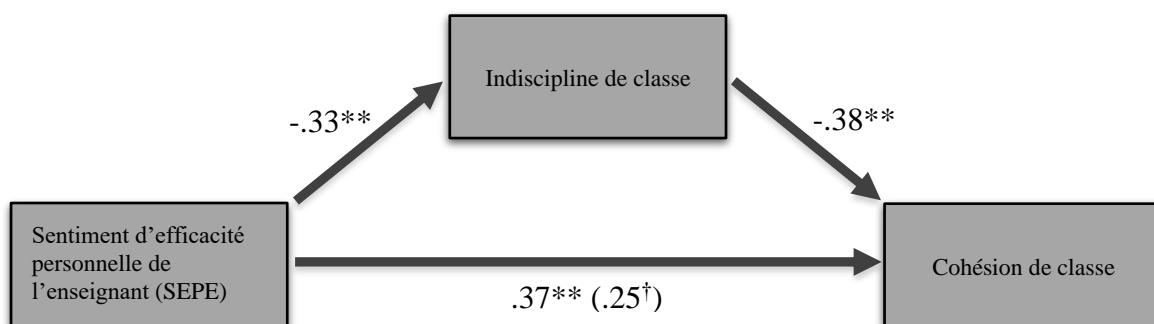
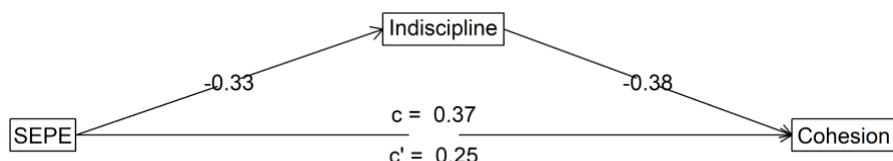


Figure 1. Coefficients de régression non standardisés de la relation entre le sentiment d'efficacité personnelle de l'enseignant (SEPE) et la cohésion de classe médié par l'indiscipline de classe. Le coefficient de régression standardisé entre le SEPE et la cohésion de classe, contrôlé par l'indiscipline de classe, est entre parenthèse.  
†  $p < .10$ , \*  $p < .05$ , \*\*  $p < .01$

Si vous souhaitez vous assurer d'avoir inscrit correctement les différents coefficients de régression standardisés, la commande ci-dessous peut vous être utile<sup>1</sup>.

```
psych::mediate(Cohesion ~ SEPE + (Indiscipline), data=Database)
```



<sup>1</sup> Notez que la fonction `mediate` de cette commande n'est pas celle du package `{mediation}`, mais celle du package `{psych}`. Pour cette raison, est développée une alternative permettant de spécifier un package précis d'une fonction grâce au « :: ». Pour l'entièreté du chapitre, n'activez que le package `{mediation}`.

Pour obtenir un résumé de toutes les étapes précédemment effectuées grâce aux différents modèles de régression, la fonction `summary` peut s'ajouter à la commande développée ci-dessus.

```
summary(psych::mediate(Cohesion~SEPE+(Indiscipline), data=Database))
```

```
Call: psych::mediate(y = Cohesion ~ SEPE + (Indiscipline), data = Database)

Total effect estimates (c)
  Cohesion   se     t df    Prob
SEPE      0.37 0.14 2.76 78 0.00711

Direct effect estimates (c')
  Cohesion   se     t df    Prob
SEPE        0.25 0.13 1.83 77 0.07160
Indiscipline -0.38 0.12 -3.08 77 0.00283

R = 0.44 R2 = 0.19   F = 9 on 2 and 77 DF   p-value: 0.000308

'a' effect estimates
  Indiscipline   se     t df    Prob
SEPE        -0.33 0.12 -2.84 78 0.00572

'b' effect estimates
  Cohesion   se     t df    Prob
Indiscipline -0.38 0.12 -3.08 77 0.00283

'ab' effect estimates
  Cohesion boot sd lower upper
SEPE       0.13 0.12 0.06  0.03  0.25
```

**Notation :** « Une analyse de médiation a été effectuée selon les étapes de Baron et Kenny (1986) afin de vérifier si le lien entre sentiment d'efficacité personnelle de l'enseignant (SEPE) et cohésion de classe est médié par l'indiscipline. Les résultats indiquent que la médiation de l'indiscipline est complète sur ce lien. Comme l'illustre la Figure 1, le coefficient de régression entre le SEPE et la cohésion est significatif,  $\beta_{\text{non standardisé}} = .37, t = 2.76, p < .01$ . Ce lien diminue et devient non significatif en contrôlant l'indiscipline en classe,  $\beta_{\text{non standardisé}} = .25, t = 1.83, p > .05$  ».

#### 4.4.9 Régression logistique

Fonctions et packages nécessaires : `glm {stats}` ; `allEffects {effects}` ; `exp {base}` ; `pR2 {pscl}` ; `odds.ratio {questionr}` ; `vif {car}`

La régression logistique permet de prédire une **variable binomiale** par des variables quantitatives ou nominales (VI). Contrairement à la régression linéaire qui permet de prédire une variable quantitative (p. ex.: nombre d'erreurs, temps de réaction), nous souhaitons connaître le risque ou la probabilité d'un événement représenté par notre variable binomiale (p. ex.: réussite ou échec, guérison ou non, maladie ou non, etc.). Cette régression est souvent utilisée en épidémiologie pour mesurer l'effet d'un facteur de risque (p. ex.: nombre de cigarettes quotidiennes) ou d'un facteur protecteur (p. ex.: heures de sport quotidiennes) sur la survenue d'une maladie (p. ex.: cancer du poumon). Il est également possible de mesurer l'effet de facteurs (p. ex. : heures de sommeil, heures de révision, l'âge) sur la réussite ou non d'un examen : on parlera là non d'un facteur « de risque », ou « protecteur », mais un facteur de « réussite ».

Exemple : Nous souhaitons connaître l'effet de différents prédicteurs, tels que l'âge (`age`), la classe d'embarquement (1<sup>ère</sup>, 2<sup>ème</sup> ou 3<sup>ème</sup> classe, `pclass`) et le sexe (`sex`) sur la survie ou non des passagers et passagères (`survived`) lors du naufrage du Titanic, variables tirées de la base de données `Titanic`. Nous avons donc **trois prédicteurs** (âge, classe et sexe) et **un événement** (survie).

**Tout d'abord**, nous devons nous assurer que les prédicteurs nominaux sont considérés en tant que facteurs par R. Dans le cas contraire, ils seront considérés comme des variables ordinaires ou quantitatives et les résultats seront erronés.

```
Titanic$pclass <- factor(Titanic$pclass)
Titanic$survived <- factor(Titanic$survived)
```

L'objet `family` nous permet de spécifier le type de régression désirée par la fonction `glm`. En l'occurrence, nous souhaitons un modèle de type binomial utilisé pour une régression logistique. La fonction `summary` nous permet d'explorer les résultats.

```
GLM_Survival <- glm(survived ~ age + pclass + sex, data=Titanic,
family=binomial(link=logit))
summary(GLM_Survival)
```

```
Call:
glm(formula = survived ~ age + pclass + sex, family = binomial(link = logit),
     data = Titanic)

Deviance Residuals:
    Min      1Q  Median      3Q      Max 
-2.6399 -0.6979 -0.4336  0.6688  2.3964 

Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) 3.522075  0.326702 10.781 < 2e-16 ***
age        -0.034393  0.006331 -5.433 5.56e-08 ***
pclass2     -1.280571  0.225538 -5.678 1.36e-08 ***
pclass3     -2.289661  0.225802 -10.140 < 2e-16 ***
sexmale     -2.497845  0.166037 -15.044 < 2e-16 *** 
---
Signif. codes:  0 ‘****’ 0.001 ‘***’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

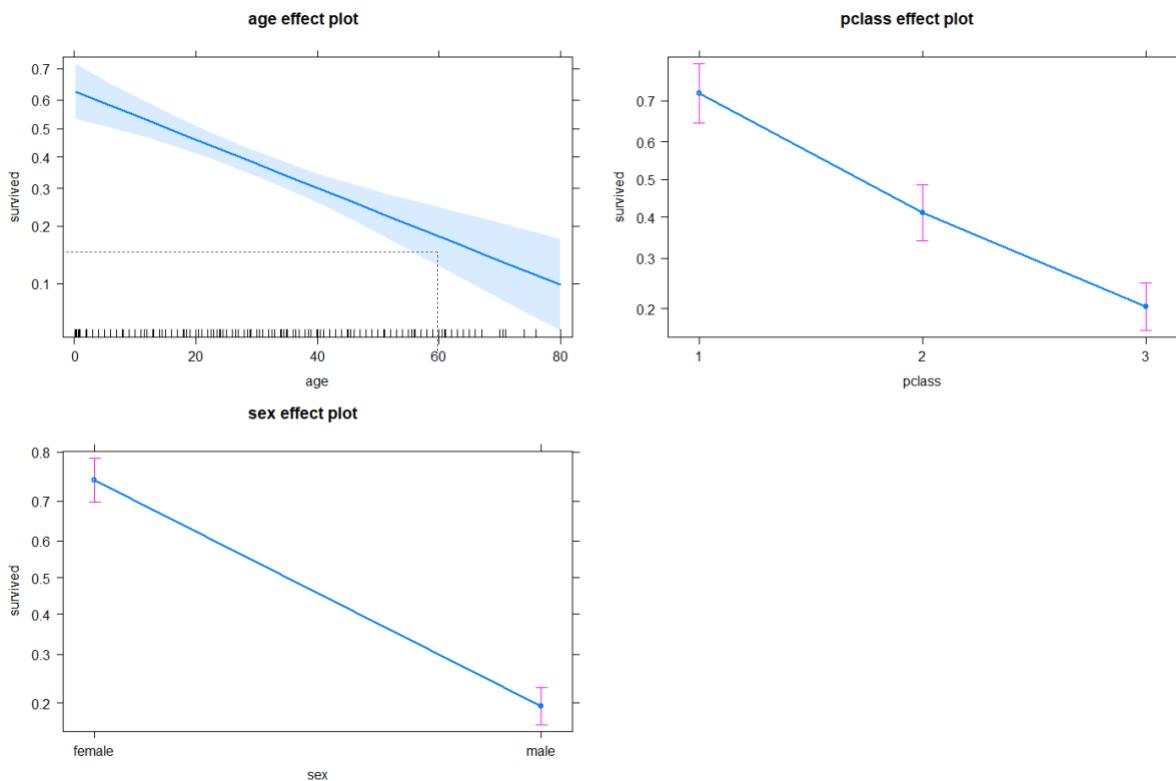
(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1414.62  on 1045  degrees of freedom
Residual deviance: 982.45  on 1041  degrees of freedom
(263 observations deleted due to missingness)
AIC: 992.45

Number of Fisher Scoring iterations: 4
```

Au vu de ces **premiers résultats**, nous constatons que tous les prédicteurs insérés ainsi que chaque niveau des variables nominales prédictrices ont une influence significative sur notre variable binomiale (*survie ou non*). Grâce à la fonction suivante, il est tout à fait possible d'illustrer ces résultats et les proportions de survie associées aux valeurs de nos prédicteurs :

```
plot(allEffects(GLM_Survival))
```



Les **graphiques** permettent de mieux visualiser l'effet des différentes prédicteurs, ainsi que la variabilité par rapport à la droite de régression. En d'autres termes, plus l'âge des passagers augmente, plus la proportion de survie diminue (par exemple, si nous « piochons » aléatoirement un passager de 60 ans, nous pouvons dire avec une certitude d'environ 80% qu'il fait partie des naufragés). Concernant la classe d'embarquement, nous constatons que plus la classe augmente, plus la proportion de survie diminue. Nous remarquons que la 1<sup>ère</sup> classe a la proportion de survie la plus élevée. Les barres d'erreur représentent les intervalles de confiance à 95%.

Le **Odds Ratio (OR)**, aussi appelé **rapport des chances** ou **risque relatif rapproché**, permet de mesurer l'effet de nos facteurs. Il se définit comme le rapport entre un groupe A (p. ex.: rétablissement avec médicament) et un groupe B (p. ex.: rétablissement sans médicament).

- Si le rapport est inférieur à 1, l'événement est moins fréquent dans le groupe A que dans le groupe B.
- Si le rapport est proche de 1, les deux événements sont indépendants.
- Si le rapport est supérieur à 1, l'événement est plus fréquent dans le groupe A que dans le groupe B.

Dans notre cas, il nous permettra de connaître les rapports de l'événement sur nos différents groupes (*femme* vs. *homme* ; 1<sup>ère</sup> classe vs. 2<sup>ème</sup> classe vs. 3<sup>ème</sup> classe ; *âge* par *âge*).

```
odds.ratio(GLM_Survival)
```

	OR	2.5 %	97.5 %	p
(Intercept)	33.854590	18.114770	65.2745	< 2.2e-16 ***
age	0.966192	0.954105	0.9781	5.557e-08 ***
pclass2	0.277879	0.177634	0.4304	1.364e-08 ***
pclass3	0.101301	0.064531	0.1565	< 2.2e-16 ***
sexmale	0.082262	0.059068	0.1133	< 2.2e-16 ***
---				
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1				

En vert, les *OR* de chacun de nos prédicteurs et en rouge, les intervalles de confiance (ici de 95%). Les intervalles de confiance nous permettent de savoir si la valeur  $OR = 1$  (qui représente l'indifférence) y est incluse. En l'occurrence, avec une certitude de 95%, les valeurs sont toutes inférieures à 1. Prenons l'âge : pour chaque année supplémentaire la fréquence de survie diminue significativement. Pour le sexe, le taux de survie des femmes est plus de 12 fois plus élevé que celui des hommes<sup>1</sup>.

Les analyses précédentes testent l'influence de chaque variable sur l'événement indépendamment de l'influence des autres. La fonction suivante permet de **tester la différence entre la variance du modèle nul** (n'incluant aucun prédicteur et représentée par la ligne NULL) et la variance résiduelle de notre modèle (incluant nos prédicteurs), prédicteur par prédicteur. L'analyse permet de savoir si chacun de nos prédicteurs diminue significativement la variance résiduelle comparativement au modèle nul. En somme, plus notre modèle explique de variance, meilleur sera-t-il.

```
anova(GLM_Survival, test="Chisq")
```

Analysis of Deviance Table						
	Model: binomial, link: logit					
	Response: survived					
	Terms added sequentially (first to last)					
Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)		
NULL		1045	1414.62			
age	1	3.238	1044	1411.38	0.07196 .	
pclass	2	155.690	1042	1255.69	< 2e-16 ***	
sex	1	273.239	1041	982.45	< 2e-16 ***	
 ---						
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1						

Ces résultats peignent une conclusion un peu différente de la dernière. Cette fois-ci, ajouter l'âge dans notre modèle de prédiction n'explique pas significativement plus de variance que le modèle nul. Cependant, la classe d'embarquement et le sexe explique significativement plus de variance que le modèle nul.

Nous souhaitons également connaître la **taille d'effet** de notre modèle ou le **model fit**. Comme aucune équivalence au  $R^2$  de la régression linéaire n'existe, nous pouvons utiliser le  $R^2$  de McFadden.

```
PR2(GLM_Survival)
```

1lh	1lhNull	G2	McFadden	r2ML	r2CU
-491.2265797	-870.5121915	758.5712236	0.4357040	0.5157779	0.6362075

La condition d'application de la multicolinéarité doit être vérifiée. Il est possible de le vérifier en exécutant des corrélations simples entre chacun de nos prédicteurs et de veiller à ce que les corrélations ne soient pas significatives. Cependant, des corrélations simples, bien que faibles, peuvent créer une dépendance biaisant nos résultats. Un indice utile à cela est l'indice de **Variation Inflation Factor (VIF)**.

<sup>1</sup> Le rapport a été calculé en divisant 1 par l'*OR*. En l'occurrence, comme l'*OR* se rapporte aux hommes :  $1/OR_{hommes} = 1/0.082 \approx 12 (= OR_{femmes})$ .

- Un indice VIF d'une valeur de 1 signifie aucune corrélation entre prédicteurs.
- Un indice VIF supérieur à la valeur de 4 implique une certaine vigilance à l'égard des résultats.
- Un indice VIF supérieur à la valeur de 10 représente une sévère multicolinéarité.

```
vif(GLM_Survival)
```

	GVIF	Df	GVIF^(1/(2*Df))
age	1.354170	1	1.163688
pclass	1.414641	2	1.090590
sex	1.052059	1	1.025699

L'indice du VIF est inscrit GVIF comme nous utilisons un modèle linéaire général (glm). Chaque indice est proche de 1, ce qui représente corrélation quasi-inexistante entre prédicteurs.

**Notation :** « Une analyse de régression logistique a été menée afin de tester l'effet de l'âge, de la classe d'embarquement et du sexe dans la survie ou non au naufrage du Titanic. Tous les prédicteurs ont un effet sur la survenue de l'événement. L'âge prédit significativement la survie,  $b = -.03$ ,  $SE_b = .01$ ,  $OR = .97$  [95%CI: .95 – .98]. La 2<sup>ème</sup> classe, comparativement à la 1<sup>ère</sup> classe, prédit significativement la survie,  $b = -1.28$ ,  $SE_b = .23$ ,  $OR = .28$  [95%CI: .17 – .43]. La 3<sup>ème</sup> classe, comparativement à la 2<sup>ème</sup> classe, prédit significativement la survie,  $b = -2.29$ ,  $SE_b = .23$ ,  $OR = .10$  [95%CI: .06 – .16]. Le genre masculin, comparativement au genre féminin, prédit significativement la survie,  $b = -2.50$ ,  $SE_b = .01$ ,  $OR = .08$  [95%CI: .06 – .11].

#### 4.4.10 Analyse en Composantes Principales (ACP)

Fonctions et packages nécessaires : KMO, principal {psych} ; bart\_spher {REdaS} ; prcomp {stats} ; get\_eig, fviz\_eig {factoextra} ; obimin {GPArotation}

L'Analyse en Composantes Principales (ou ACP) a pour but de réduire le nombre de variables quantitatives (p. ex. les différents items d'un questionnaire) en un nombre plus réduit de dimensions formées par des variables corrélées entre elles. C'est un cas particulier d'Analyse Factorielle.

Exemple : Nous aimerais savoir si les problèmes familiaux (pb\_fam), scolaires (pb\_sco), médicaux (pb\_med), sociaux (pb\_soc) et légaux (pb\_leg) rencontrés par des adolescent·es peuvent être réduits en un plus petit nombre de dimensions (appelées facteurs), variables tirées de la base de données Database.

Nous allons d'abord sélectionner les variables d'intérêt (soit les cinq citées précédemment) de la base de données Database dans une nouvelle base de données réduite (DatabaseACP).

```
DatabaseACP <- Database[c("pb_med", "pb_sco", "pb_soc", "pb_fam", "pb_leg")]
```

Deux conditions d'application doivent être vérifiées afin de savoir si une ACP est pertinente en fonction des données : l'indice KMO (Kaiser-Meyer-Olkin) et le test de sphéricité de Barlett.

**L'indice KMO** peut être considéré suffisant à partir de .5. A partir de .7, on considère l'indice comme bon, à partir de .8, il est très bon et à partir de .9, il est excellent.

```
KMO(DatabaseACP)
```

```
Kaiser-Meyer-Olkin factor adequacy
Call: KMO(r = DonneesProb)
Overall MSA = 0.63
MSA for each item =
pb_med pb_sco pb_soc pb_fam pb_leg
 0.51 0.74 0.70 0.66 0.55
```

Dans cette situation, nous observons une valeur de l'indice de KMO égal à .63 (sur la ligne spécifiée Overall MSA<sup>1</sup>), ce qui indique une valeur seulement suffisante.

**Le test de sphéricité de Bartlett** teste l'hypothèse nulle selon laquelle la matrice des corrélations est sphérique. Ce test doit être **significatif** ( $p < .05$ ) et précise si certaines variables sont suffisamment corrélées entre elles pour effectuer une ACP.

```
bart_spher(DatabaseACP, use="complete.obs")
```

```
Bartlett's Test of Sphericity
Call: bart_spher(x = DonneesProb)

X2 = 176.633
df = 10
p-value < 2.22e-16
```

Le test étant significatif, cette condition d'application soutient la pertinence de l'ACP. L'hypothèse nulle selon laquelle les corrélations entre les variables sont nulles peut être rejetée.

### Combien de composantes doivent être retenues ?

Après avoir vérifié les conditions d'application précédentes, nous pouvons passer à la **1<sup>ère</sup> étape de l'ACP**, soit **décider du nombre de composantes** à retenir. Pour cela, la commande suivante doit être compilée. Veillez à n'avoir aucune valeur manquante (ou NA), ou enlevez-les (cf. chapitre 4.2.12).

```
ACP_test <- prcomp(DatabaseACP, scale=TRUE)
```

Pour connaître le nombre de facteurs devant être retenus, il existe trois critères possibles : le critère de Kaiser-Guttman (basé sur les valeurs propres ou *eigenvalues*), le critère de Joliffe (basé sur le pourcentage de variance cumulée) et le critère de Cattell (basé sur le *scree plot*). Le critère de Kaiser-Guttman est sans doute le plus utilisé des trois.

### Critère 1 : Critère de Kaiser-Guttman (valeur propre ( $\lambda$ ) supérieure à 1)

Ce critère recommande de ne prendre en compte que les dimensions (ou facteurs) dont la valeur propre est supérieure à 1. La valeur propre représente, en quelque sorte, le nombre de variables que la dimension remplace. Ainsi, une valeur propre inférieure à 1 signifie que cette dimension explique moins de variance qu'une variable de départ, d'où l'inutilité de cette dimension.

```
get_eig(ACP_test)
```

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	2.5243500	50.48700	50.48700
Dim.2	1.3545201	27.090402	77.57740
Dim.3	0.5196035	10.392070	87.96947
Dim.4	0.3445680	6.891360	94.86083
Dim.5	0.2569584	5.139168	100.00000

<sup>1</sup> L'indice *MSA* est l'abréviation de *Measure of Sampling Adequacy*, soit la mesure de l'adéquation de l'échantillonnage. Les valeurs disposées en bas de la sortie sont les corrélations « générales » des différentes variables (*MSAi* pour *item*). Si l'on fait la moyenne de ces valeurs, nous retrouvons la valeur du KMO.

Nous pouvons constater que le critère de Kaiser-Guttman recommande de garder 2 composantes.

### **Critère 2 : Critère de Joliffe (pourcentage de variance cumulée supérieure à 75%)**

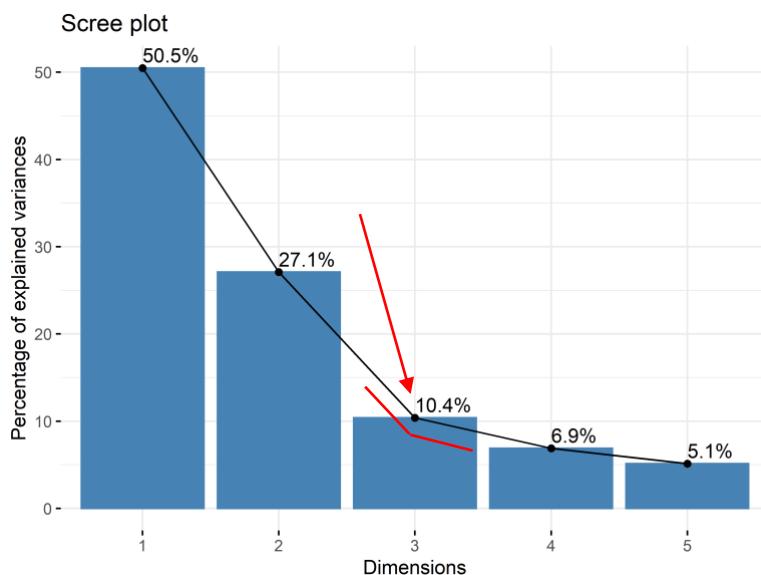
Ce critère recommande de ne prendre en compte que les dimensions dont le pourcentage de variance cumulée est supérieur à 75%. La commande à compiler est celle développée au critère précédent. Le critère de Joliffe ainsi que le critère de Kaiser-Guttman sont convergents. Selon ces deux critères, nous pouvons garder 2 composantes.

### **Critère 3 : Critère de Cattell (point d'inflexion, « coude » représenté par le scree plot)**

Ce critère recommande de ne prendre en compte que les dimensions se trouvant avant le « coude » (point d'inflexion, minimum local de la pente de la tangente) en se référant au *scree plot*. Ce graphique se base sur les variances expliquées de chaque dimension prise individuellement *avant rotation*. Ce critère n'est pas toujours simple à identifier lorsque le nombre de variables considérées dans l'ACP est élevé (il peut y avoir plusieurs coudes et plusieurs solutions).

```
fviz_eig(ACP_test, addlabels=TRUE)
```

Dans le graphique ci-dessous, nous observons un point d'inflexion, ou coude, à la 3<sup>ème</sup> dimension. Cela signifie que qu'un « pic » de la diminution de variance expliquée est atteint à la 3<sup>ème</sup> dimension et qu'ainsi, il n'est intéressant que de considérer deux dimensions.



**Pour conclure**, tous les critères convergent donc à une analyse à **2 facteurs** (à indiquer sous `nFactors`). Sous `rotate`, vous pouvez choisir une méthode de rotation des facteurs (confirmatoire, orthogonale = `varimax` ; exploratoire, oblique = `oblimin` ou `promax`). Pour ne demander aucune rotation (ce qui vous donnera des facteurs d'importance fortement décroissante), précisez `none`.

Situation 1 : Nous souhaitons effectuer une **rotation orthogonale Varimax** (facteurs non corrélés)

```
ACP_2factVAR <- principal(DatabaseACP, nfactors=2,
                           rotate="varimax") ; print(ACP_2factVAR)
```

```

> print(ACP_2facteurs)
Principal Components Analysis
Call: principal(r = DonneesProb, nfactors = 2, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
   RC1    RC2      h2    u2 com
pb_med 0.05 0.92 0.84 0.16 1.0
pb_sco 0.80 0.17 0.66 0.34 1.1
pb_soc 0.87 0.13 0.78 0.22 1.0
pb_fam 0.87 0.04 0.76 0.24 1.0
pb_leg 0.22 0.89 0.84 0.16 1.1

   RC1    RC2
SS loadings 2.20 1.68
Proportion Var 0.44 0.34
Cumulative Var 0.44 0.78
Proportion Explained 0.57 0.43
Cumulative Proportion 0.57 1.00

Mean item complexity = 1.1
Test of the hypothesis that 2 components are sufficient.

The root mean square of the residuals (RMSR) is 0.1
with the empirical chi square 18.01 with prob < 2.2e-05

Fit based upon off diagonal values = 0.95

```

La colonne « h2 » vous indique les **communautés** (ou *communalities*). Elles représentent la proportion de la variance de chaque variable expliquée par les dimensions retenues. Par exemple, 84% de la variance des problèmes médicaux (« pb\_med ») est expliquée par notre solution factorielle à deux dimensions, ce qui demeure très bon. La ligne « Proportion Var » vous indique **la variance expliquée de chacune des dimensions après rotation**. En l'occurrence, avec la rotation Varimax, la variance de chaque facteur a été maximisée.

Pour comparer les résultats *après* rotation à ceux que nous avions *avant* rotation, soit par le biais des matrices de saturations, soit par le biais du graphique des composantes, vous devrez créer une nouvelle variable ACP, mais sans rotation.

```

ACP_2factNONE <- principal(DatabaseACP, nfactors=2,
                           rotate="none")

```

Afin d'interpréter les facteurs (ou composantes) retenus, vous devez demander les **matrices de saturation**. En rajoutant un `cutoff`, les valeurs inférieures à une valeur données (ici de 0.4), seront retirées du tableau. Ceci facilite l'interprétation (surtout quand vous travaillez avec une grande base de données). Aucun `cutoff` n'est demandé pour la matrice *avant* rotation.

```

print(loadings(ACP_2factNONE)) # Avant rotation (sans cutoff)
print(loadings(ACP_2factVAR), cutoff=0.4) # Apres rotation

```

Loadings:	
	PC1 PC2
pb_med	0.523 0.755
pb_sco	0.767 -0.276
pb_soc	0.810 -0.345
pb_fam	0.758 -0.426
pb_leg	0.658 0.638

	PC1 PC2
SS loadings	2.524 1.355
Proportion Var	0.505 0.271
Cumulative Var	0.505 0.776

Loadings:	
	RC1 RC2
pb_med	0.917
pb_sco	0.798
pb_soc	0.870
pb_fam	0.869
pb_leg	0.889

	RC1 RC2
SS loadings	2.20 1.679
Proportion Var	0.44 0.336
Cumulative Var	0.44 0.776

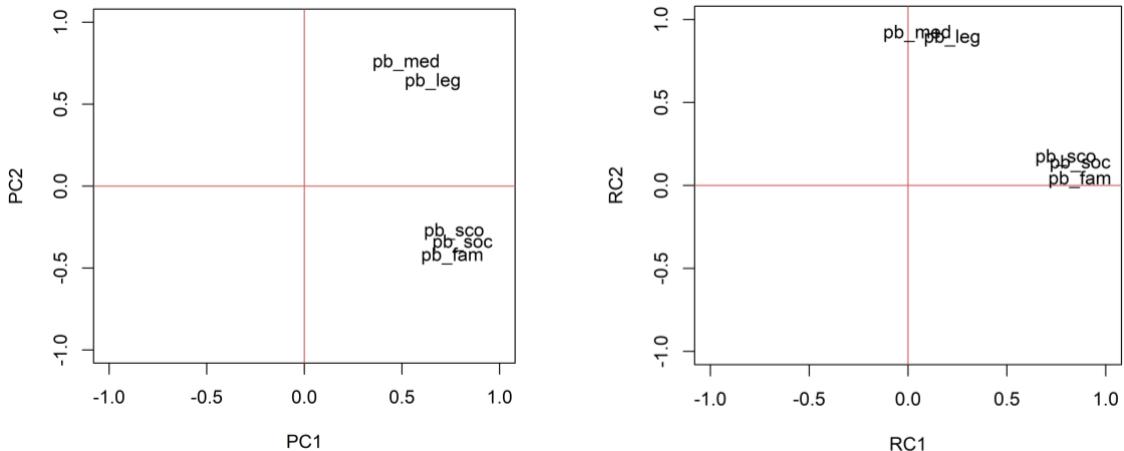
Nous observons *avant* rotation (à gauche) que les saturations sont élevées partout sans forte distinction, surtout sur le premier facteur. En revanche, *après* rotation (à droite), les saturations sont préférentiellement élevées sur une seule composante à la fois.

Pour illustrer graphiquement les saturations obtenues précédemment, nous pouvons créer un **graphique des composantes** (ou **components plot**) avant et après rotation. Cela permet d'observer l'effet que la rotation a eu sur les variables. Nous pouvons donc créer un premier graphique des composantes avant rotation.

```
plot(ACP_2factNONE$loadings, type="n", xlim=c(-1,1), ylim=c(-1,1))
abline(v=0, col="indianred") # Ajout Ligne verticale
abline(h=0, col="indianred") # Ajout Ligne horizontale
text(ACP_2factNONE$loadings, labels=names(DatabaseACP), cex=1)
```

Ainsi qu'un tableau des composantes après rotation.

```
plot(ACP_2factVAR$loadings, type="n", xlim=c(-1,1), ylim=c(-1,1))
abline(v=0, col="indianred") # Ajout Ligne verticale
abline(h=0, col="indianred") # Ajout Ligne horizontale
text(ACP_2factVAR$loadings, labels=names(DatabaseACP), cex=1)
```



**Interprétation :** Les axes représentent les composantes, et les valeurs représentent les saturations. Ainsi, chaque variable peut être située en fonction de ses saturations sur ces deux composantes. Dans le graphique avant rotation (à gauche), nous pouvons observer que chaque variable sature sur les deux composantes (ou axes). Ce qui représente chacune des composantes est alors difficile à déterminer, car elles ne sont pas associées à un groupe de variables particulier. Dans le graphique après rotation (à droite), nous constatons que les axes ont subi une rotation, ce qui permet d'associer chaque composante à un groupe de variables bien défini.

**Remarque :** Lorsque votre solution factorielle comprend plus de deux composantes, l'interprétation du graphique de saturation devient plus complexe. Dans le cas de 3 composantes retenues, il serait possible d'imaginer un graphique sur 3 dimensions, donc 3 axes, raison pour laquelle les variables saturant sur la troisième composante se situerait tout au centre des deux axes.

Enfin, vous pouvez demander les **scores factoriels** de chaque participant·e (soit les nouveaux scores pour chaque participant·e sur les facteurs retenus)<sup>1</sup>. Ce sont de nouvelles variables quantitatives sur lesquelles on peut faire d'autres analyses statistiques (ANOVA, régressions, etc.)

```
round(ACP_2factVAR$scores, 2) #Arrondissement a 2 decimales
```

	RC1	RC2
[1,]	0.22	2.16
[2,]	-1.64	-1.20
[3,]	0.63	-0.60
[4,]	0.33	-1.34
[5,]	0.94	1.49

Finalement, vous pouvez **enregistrer les scores factoriels de chaque participant·e** dans votre base de données initiale. De cette manière, ces scores seront consultables directement dans votre base de données.

```
Database <- cbind(Database, ACP_2factVAR$scores)
```

pb_soc Pbs sociaux	pb_fam Pbs familiaux	pb_leg Pbs légaux	RC1	RC2
5	5	9	0.22267590	2.158189039
1	1	1	-1.64381926	-1.203212593
6	6	3	0.62807164	-0.595219155
3	8	1	0.32721664	-1.337411629
7	7	9	0.94369850	1.488563705

**Notation :** « Afin d'évaluer si les problèmes rencontrés dans différents domaines par des adolescent·es sont liés entre eux, et par conséquent pourraient être résumés en un nombre plus réduit de dimensions, une analyse en composantes principales (ACP) avec rotation Varimax a été conduite sur les cinq domaines de problèmes évalués, les prérequis suivants étant remplis : ces variables sont mesurées sur des échelles quantitatives, le nombre total de sujets est plus de cinq fois supérieur au nombre de variables (99 sujets pour 5 variables) et l'analyse de la normalité uni- et bivariée s'est montrée satisfaisante ; de plus, 6 corrélations moyennes à fortes, toutes significatives, ont été observées entre les 5 variables. Le test de Bartlett significatif et un KMO minimal confirment la pertinence d'une ACP.

Une solution à 2 composantes a été retenue selon le critère de Kaiser-Guttman (valeur-propres > 1), Joliffe (. La première composante explique 44% de la variance totale et regroupe les problèmes scolaires, sociaux et familiaux (avec des saturations respectives après rotation de .80, .87 et .87). L'appellation

« problèmes relationnels » a été retenue dans la mesure où ce type de difficultés semble être le noyau des problèmes dans ces différents domaines. La deuxième composante explique 34% de la variance totale et regroupe les problèmes médicaux et légaux (avec des saturations respectives après rotation de .92 et .89), nous la nommons donc « problèmes médico-légaux ». Toutes les variables sont expliquées de manière satisfaisante par la solution retenue, avec des communautés allant de .66 (problèmes scolaires) à .84 (problèmes médicaux). »

<sup>1</sup> Il est également possible de calculer manuellement ces scores factoriels. Pour cela, vous sont nécessaires les scores standardisés (ou scores z) et les coefficients de scores factoriels (ou *Standardised loadings*) obtenus grâce à la commande `print(ACP_2factVAR)` en fonction de la composante. Pour l'illustrer, l'équation pour obtenir les scores factoriels de la composante 2 serait la suivante. Les  $Z_{var}$  représentent les scores standardisés de chaque participant·e :  $Scorefact\_C2 = .92 * Z_{pb\_med} + .17 * Z_{pb\_sco} + .13 * Z_{pb\_soc} + .04 * Z_{pb\_fam} + .84 * Z_{pb\_leg}$

Situation 2 : Nous souhaitons effectuer une **rotation oblique (Oblimin ou Promax)**.

Les fonctions R sont les mêmes (à part `rotate="oblimin"` ou `"promax"`), en résumé :

```
ACP_2factOBL <- principal(DatabaseACP, nfactors=2,
                           rotate="oblimin") ; print(ACP_2factOBL)
print(loadings(ACP_2factOBL), cutoff=0.4)
round(ACP_2factOBL$scores, 2)
Database <- cbind(Database, ACP_2factOBL$scores)
```

Les facteurs restent identiques, mais les valeurs sont un peu différentes de la rotation précédente (Varimax). La sortie indique en plus de l'ACP avec rotation Varimax les corrélations entre les dimensions (dans le tableau "component correlations"), car la rotation n'est cette fois pas orthogonale. Avec une rotation Varimax, ce tableau ne comprendrait que des 0.

**Notation :** « Afin d'évaluer si les problèmes rencontrés dans différents domaines par des adolescent·es sont liés entre eux, et par conséquent pourraient être résumés en un nombre plus réduit de dimensions, une analyse en composantes principales (ACP) avec rotation Oblimin a été conduite sur les cinq domaines de problèmes évalués, les prérequis suivants étant remplis : ces variables sont mesurées sur des échelles quantitatives, le nombre total de sujets est plus de cinq fois supérieur au nombre de variables (99 sujets pour 5 variables) et l'analyse de la normalité uni- et bivariée s'est montrée satisfaisante ; de plus, 6 corrélations moyennes à fortes, toutes significatives, ont été observées entre les 5 variables. Le test de Bartlett est significatif et le KMO satisfaisant.

Une solution à 2 composantes a été retenue selon le critère de Kaiser-Guttman (valeur-propres  $> 1$ ), Joliffe (pourcentage de variance cumulée supérieure à 75%). La première composante explique 44% de la variance totale et regroupe les problèmes scolaires, sociaux et familiaux (avec des saturations respectives après rotation de .79, .87 et .89). L'appellation « problèmes relationnels » a été retenue dans la mesure où ce type de difficultés semble être le noyau des problèmes dans ces différents domaines. La deuxième composante explique 34% de la variance totale et regroupe les problèmes médicaux et légaux (avec des saturations respectives après rotation de .94 et .89). L'appellation « problèmes médico-légaux » a été retenue. Les deux facteurs sont modérément corrélés positivement,  $r = .26$ . Toutes les variables sont expliquées de manière satisfaisante par la solution retenue, avec des communautés allant de .66 (problèmes scolaires) à .84 (problèmes médicaux). »

## 5. Présentation des résultats sous normes APA

Packages nécessaires : {apaTables} ; {MBESS} ; {ez}

Ce chapitre présente les commandes permettant d'obtenir des tableaux aux normes de publication selon l'*American Psychological Association* (APA). Les fonctions du packages {apaTables} permettent à la fois d'exécuter une analyse spécifique et de disposer les résultats sur un documents annexe. Les normes en cours sont la 7<sup>ème</sup> édition (ou APA 7th). La dernière version du package (2.0.5) date de l'année 2018, soit celle de l'APA 6th. Néanmoins, les deux dernières éditions maintiennent les mêmes normes de disposition de graphiques et de tableaux.

**Remarque :** Les documents sortants sont en **anglais**. Il n'existe actuellement aucune possibilité d'une sortie en français. Cependant, les documents sortants peuvent être modifiés à souhait, et ainsi traduits à votre convenance.

Plusieurs paramètres sont importants à considérer pour obtenir un document avec le format, le nom et les indices appropriés. Ci-dessous, sont présentés les paramètres généraux pour toutes les fonctions des sous-chapitres, chaque sous-chapitre ayant des paramètres spécifiques supplémentaires qui y seront présentés.

- **Base de données :** Sous `data=`, vous devez spécifier la base de données à partir de laquelle vos variables sont tirées.
- **Orientation du document :** *Paramètre valable uniquement pour les statistiques descriptives et analyses corrélatives.* Sous `landscape=`, vous pouvez spécifier l'orientation de votre document sortant. Si vous désirez un document en format paysage, orienté horizontalement, précisez `landscape=TRUE`. Si vous désirez un document en format portrait, orienté verticalement, précisez `landscape=FALSE`. Une orientation peut être préférée en fonction de la largeur du tableau. Dans un cas de statistiques descriptives, préférez un format portrait. Dans un cas de tableaux de corrélations avec un nombre important de variables quantitatives, préférez un format paysage.
- **Numéro de tableau (ou table) :** Sous `table.number=`, vous pouvez spécifier le numéro de votre tableau. Dans le document sortant, le numéro sera présenté « Table N » et se trouvera au-dessus du titre de tableau, fidèlement aux normes.
- **Nom du document sortant :** Sous `filename=`, vous pouvez spécifier le nom de votre document sortant. L'extension utilisée dans les prochains sous-chapitres est l'extension `*.doc`. Nous ne développerons pas les autres extensions possibles. Si vous souhaitez écraser un document sortant précédemment créé (pour modifier un paramètre), veillez à fermer l'ancien document, faute de quoi l'accès au fichier est dit « denied » (ou interdit).

## 5.1 Statistiques descriptives

Les statistiques descriptives présentent les moyennes, écart-types et les intervalles de confiance à 95% (à choix). Les fonctions à utiliser dépendent du nombre de facteurs désirés dans vos statistiques descriptives : pour un facteur, le design est celui d'une *One-way ANOVA* ; pour deux facteurs, le design est celui d'une *Two-way ANOVA*. Ces appellations ne signifient pas qu'une ANOVA est effectuée, mais simplement qu'elles permettent de présenter leurs résultats sous statistiques descriptives.

Ci-dessous, sont présentés les paramètres spécifiques aux fonctions de statistiques descriptives. Ces paramètres s'ajoutent aux paramètres généraux présentés précédemment.

- **Variable indépendante (facteur)** : Sous `iv=`, vous devez spécifier votre variable indépendante (ou facteur). Votre variable indépendante doit avoir été au préalablement factorisée et les modalités renommées. Dans un design two-way, deux facteurs (sous `iv1=` et `iv2=`) sont à indiquer.
- **Variable dépendante (mesure)** : Sous `dv=`, vous devez spécifier votre variable dépendante (ou mesure).
- **Intervalle de confiance (IC) à 95%** : Sous `show.conf.interval=`, vous pouvez spécifier la présence (TRUE) ou non (FALSE) d'un IC à 95% de la moyenne. Cet IC permet de connaître l'étendue dans laquelle serait la moyenne de la population observée à partir de votre échantillon avec une certitude de 95%. Les journaux scientifiques recommandent de présenter l'IC à 95% dans les articles.

### 5.1.1 Design One-way ANOVA : 1 facteur

Exemple : Nous souhaitons un tableau de statistiques descriptives (moyennes, écart-types et intervalles de confiance à 95%) des scores de perceptions illusoires de formes (ou *Illusory Pattern Perception (IPP)*, `IPP_Scores`) en fonction du genre des participant·es (`Gender`), variables tirées de la base de données `Database`. Nous souhaitons un document sortant en format portrait. Ce tableau est le 3<sup>ème</sup> de notre article.

```
apa.1way.table(iv=Gender, dv=IPP_Scores, data=Database,
                 show.conf.interval = TRUE,
                 landscape = FALSE,
                 table.number = 3,
                 filename = "IPP_by_Gender.doc")
```

Table 3

Descriptive statistics for IPP\_Scores as a function of Gender.

Gender	M	M_95%_CI	SD
Females	3.03	[2.84, 3.22]	0.66
Males	2.70	[1.76, 3.64]	0.90

Note. M and SD represent mean and standard deviation, respectively.  
LL and UL indicate the lower and upper limits of the 95% confidence interval for the mean, respectively.  
The confidence interval is a plausible range of population means that could have caused a sample mean (Cumming, 2014).

La console nous offre un aperçu du document sortant. Nous pouvons noter que le numéro, le titre et une note en fin de tableau sont inscrits, ainsi que tous les indices nécessaires à la présentation de statistiques descriptives. Dans votre dossier de projet, vous trouverez le document sortant contenant le tableau que vous pourrez insérer dans votre article.

Table 3

*Descriptive statistics for Illusory Pattern Perception scores as a function of Gender.*

Gender	<i>M</i>	<i>M</i>	
		95% CI [LL, UL]	<i>SD</i>
Females	3.03	[2.84, 3.22]	0.66
Males	2.70	[1.76, 3.64]	0.90

*Note.* *M* and *SD* represent mean and standard deviation, respectively. *LL* and *UL* indicate the lower and upper limits of the 95% confidence interval for the mean, respectively. The confidence interval is a plausible range of population means that could have created a sample mean (Cumming, 2014).

A l'ouverture du document, nous obtenons un tableau presque identique à ce que la console a présenté. Vous devrez probablement recalibrer la largeur du titre et de la note afin qu'elles soient de même largeur que le tableau. Vous pouvez également inscrire les noms complets de vos variables afin d'offrir une meilleure lisibilité à vos résultats.

### 5.1.2 Design Two-way ANOVA : 2 facteurs

Exemple : Nous souhaitons un tableau de statistiques descriptives (moyennes, écart-types et intervalles de confiance à 95%) des temps de réponse médians au *Boston Naming Test (BNT ; RespTime)* en fonction du genre des participant·es (*Gender*) et du groupe d'âge (*GrAge*), variables tirées de la base de données *Database*. Nous souhaitons un document sortant en format portrait. Ce tableau est le 1<sup>er</sup> de notre article.

```
apa.2way.table(iv1=Gender, iv2=GrAge, dv=RespTime, data=Database,
                 show.conf.interval = TRUE,
                 landscape = TRUE,
                 table.number = 1,
                 filename = "RT_by_GenderGrage.doc")
```

Table 1

Means and standard deviations for RespTime as a function of a 2(Gender) X 2(GrAge) design

	M	M_95%_CI	SD
GrAge:70-80 y.o.			
Gender			
Females	1354.65	[1181.03, 1528.28]	287.32
Males	1313.23	[1198.76, 1427.70]	170.39
GrAge:81-100 y.o.			
Gender			
Females	1531.03	[1350.52, 1711.54]	338.75
Males	1575.30	[1353.89, 1796.71]	178.32

*Note.* *M* and *SD* represent mean and standard deviation, respectively.  
*LL* and *UL* indicate the lower and upper limits of the 95% confidence interval for the mean, respectively.  
The confidence interval is a plausible range of population means that could have created a sample mean (Cumming, 2014).

La console nous offre un aperçu du document sortant. Nous pouvons noter que le numéro, le titre et une note en fin de tableau sont inscrits, ainsi que tous les indices nécessaires à la présentation de statistiques descriptives. Dans votre dossier de projet, vous trouverez le document sortant contenant le tableau que vous pourrez insérer dans votre article.

Table 1

*Means and standard deviations for median BNT response time as a function of a 2(Gender) X 2(Group of age) design*

Group of age: 70 to 80 years old				
Gender	M	M	95% CI [LL, UL]	SD
Females	1354.65	[1181.03, 1528.28]	287.32	
Males	1313.23	[1198.76, 1427.70]	170.39	
Group of age: 81 to 100 years old				
Gender	M	M	95% CI [LL, UL]	SD
Females	1531.03	[1350.52, 1711.54]	338.75	
Males	1575.30	[1353.89, 1796.71]	178.32	

*Note.* M and SD represent mean and standard deviation, respectively. LL and UL indicate the lower and upper limits of the 95% confidence interval for the mean, respectively. The confidence interval is a plausible range of population means that could have created a sample mean (Cumming, 2014).

A l'ouverture du document, nous obtenons un tableau presque identique à ce que la console a présenté. Vous devrez probablement recalibrer la largeur du titre et de la note afin qu'elles soient de même largeur que le tableau. Vous pouvez également inscrire les noms complets de vos variables afin d'offrir une meilleure lisibilité à vos résultats.

## 5.2 ANOVA

Les résultats d'ANOVA présentent, entre autres, l'indice *F*, ses degrés de libertés (*df*), la valeur *p*, et la taille d'effet ( $\eta_p^2$ ). Contrairement aux statistiques descriptives, la commande d'analyse de variance peut comporter autant de facteurs que désirés.

Ci-dessous, sont présentés les paramètres spécifiques aux fonctions d'analyse de variance. Ces paramètres s'ajoutent aux paramètres généraux présentés précédemment.

- **L'analyse de variance :** Dans la commande `lm()`, vous devez spécifier l'analyse comme dans une analyse de régression linéaire, tel que `lm(VD~VI, data=Database)`. Le résultat sera ensuite utilisé dans la seconde commande.
- **Stratégie d'analyse :** Sous `type=`, vous devez spécifier la stratégie d'analyse. Si vous vous intéressez également à un effet d'interaction, spécifiez `type=3`. Si vous n'avez aucun effet d'interaction ou que vous n'avez qu'un seul facteur, notez `type=2`.
- **Intervalle de confiance (IC) de la taille d'effet :** Sous `conf.level=`, vous pouvez spécifier l'IC de la taille d'effet. De manière générale, et comme le niveau de significativité est fixé à  $p < .05$ , nous recommandons un IC de 95%, noté `conf.level=.95`. Cet IC nous permet de situer, avec une certitude de 95%, la valeur de la taille d'effet de la population observée à partir de notre échantillon. Si l'IC comprend la valeur zéro (.00), alors aucun effet significatif n'est à signaler.

### 5.2.1 ANOVA à un ou plusieurs facteurs indépendants

Les situations décrites ci-dessous reprennent les exemples développés aux précédents sous-chapitres de statistiques descriptives, respectivement de la *one-way ANOVA* et de la *two-way ANOVA* (avec et sans interaction). Dans ce sous-chapitre, seules sont développées les analyses de variances à **un ou plusieurs facteurs indépendants**. Pour des tableaux d'analyses de variance à mesures répétées ou mixtes, référez-vous au sous-chapitre suivant.

**Situation 1 :** Nous souhaitons obtenir un tableau des résultats de l'**analyse de variance à 1 facteur indépendant** des scores *IPP* (*IPP\_Scores*) en fonction du genre des participant·es (*Gender*), variables tirées de la base de données *Database*. L'IC du  $\eta^2$  est fixé à 95%. Ce tableau est le 4<sup>ème</sup> de notre article.

```
lm_output <- lm(IPP_Scores ~ Gender, data = Database)
apa.aov.table(lm_output, type = 2, conf.level = 0.95,
               table.number = 4,
               filename = "AOV-IPP_by_Gender.doc")
```

Table 4

ANOVA results using IPP\_Scores as the dependent variable

Predictor	SS	df	MS	F	p	partial_eta2	CI_95_partial_eta2
(Intercept)	451.06	1	451.06	951.04	.000		
Gender	0.60	1	0.60	1.26	.267	.02	[.00, .15]
Error	25.14	53	0.47				

Note: Values in square brackets indicate the bounds of the 95% confidence interval for partial eta-squared

La console nous offre un aperçu du document sortant. Nous pouvons noter que le numéro, le titre et une note en fin de tableau sont inscrits, ainsi que tous les indices nécessaires à la présentation de résultats d'analyse de variance. Dans votre dossier de projet, vous trouverez le document sortant contenant le tableau que vous pourrez insérer dans votre article.

Table 4

*Fixed-Effects ANOVA results using Illusory Pattern Perception scores as the criterion*

Predictor	Sum of Squares	df	Mean Square	F	p	partial $\eta^2$	partial $\eta^2$ 95% CI [LL, UL]
(Intercept)	451.06	1	451.06	951.04	.000		
Gender	0.60	1	0.60	1.26	.267	.02	[.00, .15]
Error	25.14	53	0.47				

*Note.* LL and UL represent the lower-limit and upper-limit of the partial  $\eta^2$  confidence interval, respectively.

A l'ouverture du document, nous obtenons un tableau presque identique à ce que la console a présenté. Vous devrez probablement recalibrer la largeur du titre et de la note afin qu'elles soient de même largeur que le tableau. Vous pouvez également inscrire les noms complets de vos variables afin d'offrir une meilleure lisibilité à vos résultats.

Situation 2 : Nous souhaitons obtenir un tableau des résultats de l'analyse de variance des temps de réponse médians au *BNT* (*RespTime*) en fonction du genre des participant·es (*Gender*), du groupe d'âge (*GrAge*), soit **2 facteurs indépendants avec interaction**, variables tirées de la base de données *Database*. L'IC du  $\eta_p^2$  est fixé à 95%. Ce tableau est le 2<sup>ème</sup> de notre article.

```
lm_output <- lm(RespTime ~ Gender * GrAge, data = Database)
apa.aov.table(lm_output, type = 3, conf.level = 0.95,
               table.number = 2,
               filename = "AOV-RT_by_GenderGrage.doc")
```

Table 2

ANOVA results using RespTime as the dependent variable

Predictor	SS	df	MS	F	p	partial_eta2	CI_95_partial_eta2
(Intercept)	18745208.48	1	18745208.48	29.98	.000		
Gender	28696.32	1	28696.32	0.05	.831	.00	[.00, .08]
GrAge	4227416.82	1	4227416.82	6.76	.013	.14	[.01, .33]
Gender x GrAge	243689.86	1	243689.86	0.39	.536	.01	[.00, .13]
Error	25637714.21	41	625310.10				

Note: Values in square brackets indicate the bounds of the 95% confidence interval for partial eta-squared

Table 2

Fixed-Effects ANOVA results using median BNT response time as the criterion

Predictor	Sum of Squares	df	Mean Square	F	p	partial $\eta^2$	95% CI [LL, UL]
(Intercept)	18745208.48	1	18745208.48	29.98	.000		
Gender	28696.32	1	28696.32	0.05	.831	.00	[.00, .08]
Group of Age	4227416.82	1	4227416.82	6.76	.013	.14	[.01, .33]
Gender x Group of Age	243689.86	1	243689.86	0.39	.536	.01	[.00, .13]
Error	25637714.21	41	625310.10				

Note. LL and UL represent the lower-limit and upper-limit of the partial  $\eta^2$  confidence interval, respectively.

Situation 3 : Nous souhaitons obtenir un tableau des résultats de l'analyse de variance des temps de réponse médians au *BNT* (*RespTime*) en fonction du genre des participant·es (*Gender*), du groupe d'âge (*GrAge*), soit **2 facteurs indépendants sans interaction**, variables tirées de la base de données *Database*. L'IC du  $\eta_p^2$  est fixé à 95%. Ce tableau est le 2<sup>ème</sup> de notre article.

```
lm_output <- lm(RespTime ~ Gender + GrAge, data = Database)
apa.aov.table(lm_output, type = 2, conf.level = 0.95,
               table.number = 2,
               filename = "AOV-RT_by_GenderGrageNI.doc")
```

Table 2

ANOVA results using RespTime as the dependent variable

Predictor	SS	df	MS	F	p	partial_eta2	CI_95_partial_eta2
Gender	374076.24	1	374076.24	0.61	.440	.01	
GrAge	4661049.01	1	4661049.01	7.56	.009	.15	[.01, .34]
Error	25881404.07	42	616223.91				

Note: Values in square brackets indicate the bounds of the 95% confidence interval for partial eta-squared

Table 2

*Fixed-Effects ANOVA results using median BNT responses time as the criterion*

Predictor	Sum of Squares	df	Mean Square	F	p	partial $\eta^2$	partial $\eta^2$ 95% CI [LL, UL]
Gender	374076.24	1	374076.24	0.61	.440	.01	
Group of Age	4661049.01	1	4661049.01	7.56	.009	.15	[.01, .34]
Error	25881404.07	42	616223.91				

Note. LL and UL represent the lower-limit and upper-limit of the partial  $\eta^2$  confidence interval, respectively.

### 5.2.2 ANOVA à mesures répétées et ANOVA mixtes

Les situations décrites ci-dessous reprennent les exemples développés aux chapitres d'analyse de variance à mesures répétées et mixtes. L'analyse est effectuée grâce à la commande `ezANOVA`, raison pour laquelle le package `{ez}` doit être activé.

Ci-dessous, sont présentés les paramètres spécifiques aux fonctions d'analyse de variance à mesures répétées et mixtes. Ces paramètres s'ajoutent aux paramètres généraux présentés précédemment.

- **L'analyse de variance :** Dans la commande `ezANOVA()`, vous devez spécifier l'analyse comme utilisée dans les chapitres dédiés à ces fonctions. Le résultat sera ensuite utilisé dans la seconde commande.
- **Correction de sphéricité :** Sous `correction=`, vous devez spécifier la correction ou non de la sphéricité. Si la condition de sphéricité, mesurée par le test de Mauchly, est respectée (test non significatif), alors précisez `correction="none"`. Si la condition de sphéricité n'est pas respectée (test significatif), alors précisez `correction="GG"` : le tableau de résultats prendra alors en compte la correction de Greenhouse-Geisser.
- **Titre de tableau :** Sous `table.title=`, vous pouvez spécifiez le texte désiré pour titrer votre tableau. Contrairement aux fonctions précédentes, le titre n'est pas spécifié d'office. Référez-vous aux exemples ci-dessous pour un modèle de titre.

**Situation 1 :** Nous souhaitons obtenir un tableau des résultats de l'**analyse de variance à mesures répétées** du jugement porté à des visages (`ScoreJugement`) en fonction du trait (`Traits`) et du port de lunettes (`Lunettes`), et dont la variable d'identification est `Numparticipant`, variables tirées de la base de données `Database_lg`.<sup>1</sup> Nous souhaitons une correction de la sphéricité. Ce tableau est titré et est le 10<sup>ème</sup> de notre article.

```
ez_output <- ezANOVA(data = Database_lg,
                      dv = .(ScoreJugement),
                      wid = .(Numparticipant),
                      within = .(Traits, Lunettes),
                      detailed = TRUE, type = 3)

apa.ezANOVA.table(ez_output, correction="GG",
                   table.title="Repeated Measures ANOVA using...",
                   table.number=10,
                   filename="ezAOV-Sjudg_by_TrLun.doc")
```

Table 10

Repeated Measures ANOVA using Judgment scores as the criterion

Predictor	df_num	df_den	Epsilon	SS_num	SS_den	F	p	ges
(Intercept)	1.00	102.00		17885.42	393.14	4640.37	.000	.96
Lunettes	1.00	102.00		24.88	95.78	26.50	.000	.03
Traits	2.82	287.77	0.71	251.31	215.62	118.89	.000	.24
Traits x Lunettes	3.01	306.58	0.75	68.42	109.74	63.59	.000	.08

Note. df\_num indicates degrees of freedom numerator. df\_den indicates degrees of freedom denominator. Epsilon indicates Greenhouse-Geisser multiplier for degrees of freedom, p-values and degrees of freedom in the table incorporate this correction. SS\_num indicates sum of squares numerator. SS\_den indicates sum of squares denominator. ges indicates generalized eta-squared.

La console nous offre un aperçu du document sortant. Nous pouvons noter que le numéro, le titre et une note en fin de tableau sont inscrits, ainsi que tous les indices nécessaires à la présentation de résultats d'analyse de variance. Dans votre dossier de projet, vous trouverez le document sortant contenant le tableau que vous pourrez insérer dans votre article.

Table 10

Repeated Measures ANOVA using Judgment scores as the criterion

Predictor	df <sub>Num</sub>	df <sub>Den</sub>	Epsilon	SS <sub>Num</sub>	SS <sub>Den</sub>	F	p	η <sup>2</sup> <sub>g</sub>
(Intercept)	1.00	102.00		17885.42	393.14	4640.37	.000	.96
Lunettes	1.00	102.00		24.88	95.78	26.50	.000	.03
Traits	2.82	287.77	0.71	251.31	215.62	118.89	.000	.24
Traits x Lunettes	3.01	306.58	0.75	68.42	109.74	63.59	.000	.08

Note. df<sub>Num</sub> indicates degrees of freedom numerator. df<sub>Den</sub> indicates degrees of freedom denominator. Epsilon indicates Greenhouse-Geisser multiplier for degrees of freedom, p-values and degrees of freedom in the table incorporate this correction. SS<sub>Num</sub> indicates sum of squares numerator. SS<sub>Den</sub> indicates sum of squares denominator. η<sup>2</sup><sub>g</sub> indicates generalized eta-squared.

A l'ouverture du document, nous obtenons un tableau presque identique à ce que la console a présenté. Vous devrez probablement recalibrer la largeur du titre et de la note afin qu'elles soient de même largeur que le tableau. Vous pouvez également inscrire les noms complets de vos variables afin d'offrir une meilleure lisibilité à vos résultats.

<sup>1</sup> Cet exemple est tiré du chapitre 4.4.4.4 ANOVA à mesures répétées (Repeated Measures ANOVA).

**Situation 2 :** Nous souhaitons obtenir un tableau des résultats de l'**analyse de variance mixte** du jugement porté à des visages (`ScoreJugement`) en fonction du trait (`Traits`), du port de lunettes (`Lunettes`) et de l'ethnie des visages présentés (`EthnieVisages`), et dont la variable d'identification est `Numparticipant`, variables tirées de la base de données `Database_lg`.<sup>1</sup> Nous souhaitons une correction de la sphéricité. Ce tableau est titré et est le 11<sup>ème</sup> de notre article.

```
ez_ANOVA <- ezANOVA(data = Database_lg,
                      dv = .(ScoreJugement),
                      wid = .(Numparticipant),
                      within = .(Traits, Lunettes),
                      between = .(EthnieVisages),
                      detailed = TRUE, type = 3)

apa.ezANOVA.table(ez_output, correction="GG",
                   table.title="Repeated Measures ANOVA using...",
                   table.number=11,
                   filename="ezAOV-Sjug_by_TrLunEv.doc")
```

Table 11

Repeated Measures ANOVA using Judgment scores as the criterion

Predictor	df_num	df_den	Epsilon	SS_num	SS_den	F	p	ges
(Intercept)	1.00	101.00		17829.02	391.93	4594.50	.000	.96
EthnieVisages	1.00	101.00		1.21	391.93	0.31	.578	.00
Lunettes	1.00	101.00		24.78	95.77	26.13	.000	.03
EthnieVisages x Lunettes	1.00	101.00		0.01	95.77	0.01	.928	.00
Traits	2.84	286.86	0.71	246.64	206.60	120.57	.000	.24
EthnieVisages x Traits	2.84	286.86	0.71	9.02	206.60	4.41	.006	.01
Traits x Lunettes	3.16	319.35	0.79	66.31	103.37	64.79	.000	.08
EthnieVisages x Traits x Lunettes	3.16	319.35	0.79	6.37	103.37	6.22	.000	.01

Note. df\_num indicates degrees of freedom numerator. df\_den indicates degrees of freedom denominator. Epsilon indicates Greenhouse-Geisser multiplier for degrees of freedom, p-values and degrees of freedom in the table incorporate this correction. SS\_num indicates sum of squares numerator. SS\_den indicates sum of squares denominator. ges indicates generalized eta-squared.

Table 11

Repeated Measures ANOVA using Judgment scores as the criterion

Predictor	dfNum	dfDen	Epsilon	SSNum	SSDen	F	p	eta <sup>2</sup> ges
(Intercept)	1.00	101.00		17829.02	391.93	4594.50	.000	.96
EthnieVisages	1.00	101.00		1.21	391.93	0.31	.578	.00
Lunettes	1.00	101.00		24.78	95.77	26.13	.000	.03
EthnieVisages x Lunettes	1.00	101.00		0.01	95.77	0.01	.928	.00
Traits	2.84	286.86	0.71	246.64	206.60	120.57	.000	.24
EthnieVisages x Traits	2.84	286.86	0.71	9.02	206.60	4.41	.006	.01
Traits x Lunettes	3.16	319.35	0.79	66.31	103.37	64.79	.000	.08
EthnieVisages x Traits x Lunettes	3.16	319.35	0.79	6.37	103.37	6.22	.000	.01

Note. dfNum indicates degrees of freedom numerator. dfDen indicates degrees of freedom denominator. Epsilon indicates Greenhouse-Geisser multiplier for degrees of freedom, p-values and degrees of freedom in the table incorporate this correction. SSNum indicates sum of squares numerator. SSDen indicates sum of squares denominator. η<sup>2</sup> ges indicates generalized eta-squared.

<sup>1</sup> Cet exemple est tiré du chapitre 4.4.4.5 ANOVA mixte (mesures répétées et groupes indépendants).

### 5.3 Corrélation

Les résultats d'analyses de corrélations présentent les moyennes, écart-types, coefficients et intervalles de confiance (95%). La commande présentée offre uniquement la possibilité d'effectuer une corrélation de Bravais-Pearson (test paramétrique). La commande peut comporter autant de variables quantitatives que désiré. Un tableau comportant un nombre important de variables quantitatives sera mis de préférence en format paysage.

Ci-dessous, est présenté le paramètre spécifique à la fonction d'analyse corrélationnelle. Ce paramètre s'ajoute aux paramètres généraux présentés précédemment.

- **Intervalle de confiance (IC) à 95% :** Sous `show.conf.interval=`, vous pouvez spécifier la présence (TRUE) ou non (FALSE) d'un IC à 95% du coefficient de corrélation. Cet IC permet de connaître l'étendue dans laquelle serait le coefficient de la population observée à partir de votre échantillon, en l'occurrence avec une certitude de 95%. Si la valeur zéro (.00) est comprise dans l'IC, alors le coefficient n'est pas significatif. Les journaux recommandent de présenter l'IC à 95% dans les articles.

Exemple : Nous souhaitons obtenir un tableau des coefficients de corrélation avec IC à 95% des différents types de problèmes, soit familiaux (`pb_fam`), scolaires (`pb_sco`), sociaux (`pb_soc`), légaux (`pb_leg`) et médicaux (`pb_med`), rencontrés par des adolescent·es, variables tirées de la base de données `Database`.<sup>1</sup> Nous souhaitons un document sortant au format portrait. Ce tableau est le 12<sup>ème</sup> de notre article. . Nous souhaitons une correction de la sphéricité. Ce tableau est titré et est le 10<sup>ème</sup> de notre article.

Tout comme dans les deux sous-chapitres d'analyse de variance, il est nécessaire d'effectuer une manipulation avant de lancer la commande de création de tableau. Cette manipulation reste cependant similaire au chapitre de l'analyse de corrélation (*Situation 2*) et consiste à sélectionner les variables quantitatives pour l'analyse depuis votre base de données.

```
Data <- Database[,c("pb_fam","pb_sco","pb_soc","pb_leg","pb_med")]
apa.cor.table(data = Data, show.conf.interval = TRUE,
               table.number = 12, landscape = FALSE,
               filename = "CorrPbs.doc")
```

Variable	M	SD	1	2	3	4
1. pb_fam	5.36	2.24				
2. pb_sco	4.70	2.32	.54**			
			[.38, .66]			
3. pb_soc	4.84	2.05	.67**	.57**		
			[.55, .77]	[.42, .69]		
4. pb_leg	3.96	2.36	.18	.35**	.30**	
			[-.01, .37]	[.16, .51]	[.11, .47]	
5. pb_med	3.17	2.19	.15	.14	.17	.67**
			[-.05, .34]	[-.06, .33]	[-.03, .35]	[.54, .77]

Means, standard deviations, and correlations with confidence intervals

Note. M and SD are used to represent mean and standard deviation, respectively.  
Values in square brackets indicate the 95% confidence interval.  
The confidence interval is a plausible range of population correlations  
that could have caused the sample correlation (Cumming, 2014).  
\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

<sup>1</sup> Cet exemple est tiré du chapitre 4.4.6.1 Corrélation de Bravais-Pearson (*Situation 2*).

La console nous offre un aperçu du document sortant. Nous pouvons noter que le numéro, le titre et une note en fin de tableau sont inscrits, ainsi que tous les indices nécessaires à la présentation de résultats d'analyses corrélatives. Vous remarquerez qu'aucune valeur  $p$  n'est inscrite, mais que leur significativité est symbolisée par \* (si  $.01 < p < .05$ ) et \*\* (si  $p < .01$ ). Dans votre dossier de projet, vous trouverez le document sortant contenant le tableau que vous pourrez insérer dans votre article.

Table 12

*Means, standard deviations, and correlations with confidence intervals*

Variable	<i>M</i>	<i>SD</i>	1	2	3	4
1. Family issues	5.36	2.24				
2. School issues	4.70	2.32	.54** [.38, .66]			
3. Social issues	4.84	2.05	.67** [.55, .77]	.57** [.42, .69]		
4. Legal issues	3.96	2.36	.18 [-.01, .37]	.35** [.16, .51]	.30** [.11, .47]	
5. Medical issues	3.17	2.19	.15 [-.05, .34]	.14 [-.06, .33]	.17 [-.03, .35]	.67** [.54, .77]

*Note.*  $M$  and  $SD$  are used to represent mean and standard deviation, respectively. Values in square brackets indicate the 95% confidence interval for each correlation. The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

A l'ouverture du document, nous obtenons un tableau presque identique à ce que la console a présenté. Vous devrez probablement recalibrer la largeur du titre et de la note afin qu'elles soient de même largeur que le tableau. Vous pouvez également inscrire les noms complets de vos variables afin d'offrir une meilleure lisibilité à vos résultats.

## 5.4 Régression linéaire

Les résultats de régressions linéaires présentent les coefficients non standardisés ( $b$ ), les coefficients standardisés ( $\beta$ ), les corrélations semi-partielles ( $sr^2$ ), les corrélations simples (ou *zero-order*,  $r^2$ ), les pourcentages de variance du modèle ( $R^2$ ), les changements de  $R^2$  lors de comparaisons de modèles ( $\Delta R^2$ ) et leurs intervalles de confiance. Les tableaux comparent les coefficients de corrélation simple, soit chaque prédicteur pris individuellement avec le critère, avec les coefficients standardisés du modèle : ainsi, nous pouvons savoir si le prédicteur est toujours significativement corrélé avec le critère malgré la présence de tous les autres prédicteurs. La commande peut comporter autant de modèles et de prédicteurs que désiré.

Ci-dessous, sont présentés les paramètres spécifiques à la fonction d'analyse de régression. Ces paramètres s'ajoutent aux paramètres généraux présentés précédemment.

- **Analyse de régression linéaire :** Dans `lm()`, vous devez spécifier l'analyse de régression linéaire, tel que `bloCn <- lm(VD~VI, data=Database)`. Il est possible de comparer plusieurs modèles entre eux (voir *Situation 2* et *Situation 3*) et d'obtenir un

$\Delta R^2$ . Ces différents modèles sont sauvegardés en blocs puis comparés dans la commande de tableau.

- **Intervalles de confiance (IC) des proportions de variance :** Sous le paramètre `prop.var.conf.level=`, vous pouvez spécifier l'IC de la proportion de variance. De manière générale, et comme le niveau de significativité est fixé à  $p < .05$ , nous recommandons un IC de 95%, noté `prop.var.conf.level=.95`. Cet IC nous permet de situer, avec une certitude de 95%, la valeur des proportions de variance de la population observée à partir de notre échantillon. Si l'IC comprend la valeur zéro (.00), alors aucun effet significatif n'est à signaler.

Situation 1 : Nous souhaitons obtenir un tableau de l'**analyse de régression linéaire multiple** de la prédiction de la consommation de drogue (`Conso_Dro`) chez les adolescent·es par différents types de problèmes, soit médicaux (`pb_med`), scolaires (`pb_sco`), sociaux (`pb_soc`) et familiaux (`pb_fam`), variables tirées de la base de données `Database`.<sup>1</sup> Nous souhaitons un IC de 95% pour les différentes proportions de variance. Ce tableau est le 7<sup>ème</sup> de notre article.

```
bloc01 <- lm(Conso_Dro ~ pb_med + pb_sco + pb_soc + pb_fam,
               data = Database)
apa.reg.table(bloc01,
               table.number = 7,
               prop.var.conf.level = .95,
               filename = "LM1mod-ConsoDro.doc")
```

Table 7

Regression results using Conso\_Dro as the criterion

Predictor	b	b_95%_CI	beta	beta_95%_CI	sr2	sr2_95%_CI	r	Fit
(Intercept)	3.55**	[2.76, 4.33]						
pb_med	0.32**	[0.20, 0.45]	0.43	[0.27, 0.60]	.18	[.06, .31]	.49**	
pb_sco	-0.13	[-0.27, 0.01]	-0.18	[-0.38, 0.02]	.02	[-.02, .07]	.16	
pb_soc	0.29**	[0.11, 0.48]	0.37	[0.14, 0.60]	.07	[-.01, .14]	.43**	
pb_fam	0.10	[-0.06, 0.26]	0.14	[-0.09, 0.36]	.01	[-.02, .04]	.35**	
							R2 = .391**	
							95% CI[.22,.49]	

Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also significant.  
b represents unstandardized regression weights. beta indicates the standardized regression weights.  
sr2 represents the semi-partial correlation squared. r represents the zero-order correlation.  
Square brackets are used to enclose the lower and upper limits of a confidence interval.  
\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

La console nous offre un aperçu du document sortant. Nous pouvons noter que le numéro et une note en fin de tableau sont inscrits, ainsi que tous les indices nécessaires à la présentation de résultats d'analyses de régression linéaire. Vous remarquerez qu'aucune valeur  $p$  n'est inscrite, mais que leur significativité est symbolisée par \* (si  $.01 < p < .05$ ) et \*\* (si  $p < .01$ ). Nous apprenons que les problèmes familiaux ne prédisent plus la consommation de drogue à cause de l'influence de tous les autres prédicteurs ( $r = .10$ ), alors que cela était le cas lors d'une corrélation simple ( $r = .35**$ ) : l'apport spécifique des problèmes familiaux est désormais trop faible pour être significatif. Dans votre dossier de projet, vous trouverez le document sortant contenant le tableau que vous pourrez insérer dans votre article.

<sup>1</sup> Cet exemple est tiré du chapitre 4.4.8.1 Régression linéaire multiple (et simple)

Table 7

Regression results using Drug consumption as the criterion

Predictor	<i>b</i>	<i>b</i> 95% CI [LL, UL]	<i>beta</i>	<i>beta</i> 95% CI [LL, UL]	<i>sr</i> <sup>2</sup>	<i>sr</i> <sup>2</sup> 95% CI [LL, UL]	<i>r</i>	Fit
(Intercept)	3.55**	[2.76, 4.33]						
Medical issues	0.32**	[0.20, 0.45]	0.43	[0.27, 0.60]	.18	[.06, .31]	.49**	
School issues	-0.13	[-0.27, 0.01]	-0.18	[-0.38, 0.02]	.02	[-.02, .07]	.16	
Social issues	0.29**	[0.11, 0.48]	0.37	[0.14, 0.60]	.07	[-.01, .14]	.43**	
Family issues	0.10	[-0.06, 0.26]	0.14	[-0.09, 0.36]	.01	[-.02, .04]	.35**	
								<i>R</i> <sup>2</sup> = .391** 95% CI [.22,.49]

Note. A significant *b*-weight indicates the beta-weight and semi-partial correlation are also significant. *b* represents unstandardized regression weights. *beta* indicates the standardized regression weights. *sr*<sup>2</sup> represents the semi-partial correlation squared. *r* represents the zero-order correlation. *LL* and *UL* indicate the lower and upper limits of a confidence interval, respectively.

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

A l'ouverture du document, nous obtenons un tableau presque identique à ce que la console a présenté. Vous devrez probablement recalibrer la largeur du titre et de la note afin qu'elles soient de même largeur que le tableau. Vous pouvez également inscrire les noms complets de vos variables afin d'offrir une meilleure lisibilité à vos résultats.

**Situation 2 :** Nous souhaitons obtenir un tableau de l'**analyse de régression linéaire multiple** (2 modèles hiérarchiques) de la prédiction de la consommation de drogue (*Conso\_Dro*) chez les adolescent·es par les problèmes médicaux (*pb\_med*) et sociaux (*pb\_soc*) dans un premier bloc et avec leur interaction dans un second bloc, variables tirées de la base de données *Database*.<sup>1</sup> Nous souhaitons un IC de 95% pour les différentes proportions de variance. Ce tableau est le 8<sup>ème</sup> de notre article.

```
bloc01 <- lm(Conso_Dro ~ pb_med + pb_soc,
               data = Database)
bloc02 <- lm(Conso_Dro ~ pb_med + pb_soc + I(pb_med * pb_soc),
               data = Database)
apa.reg.table(bloc01, bloc02,
               table.number = 8,
               prop.var.conf.level = .95,
               filename = "LM2modI-ConsoDro.doc")
```

<sup>1</sup> Cet exemple est tiré du sous-chapitre précédent en y incluant les deux seuls prédicteurs significatifs et leur interaction. Cependant, selon le changement de variance ( $\Delta R^2$ ) non significatif, l'interaction n'est pas pertinente.

Table 8

Regression results using Conso\_Dro as the criterion

Predictor	b	b_95%_CI	beta	beta_95%_CI	sr2	sr2_95%_CI	r	Fit	Difference
(Intercept)	3.53**	[2.80, 4.26]							
pb_med	0.32**	[0.20, 0.44]	0.43	[0.27, 0.60]	.18	[.06, .31]	.49**		
pb_soc	0.28**	[0.15, 0.42]	0.36	[0.19, 0.52]	.12	[.02, .23]	.43**		
								R2 = .366**	
								95% CI [.21,.48]	
(Intercept)	3.02**	[1.95, 4.08]							
pb_med	0.50**	[0.20, 0.80]	0.67	[0.27, 1.07]	.07	[-.01, .16]	.49**		
pb_soc	0.41**	[0.18, 0.63]	0.51	[0.23, 0.79]	.08	[-.00, .17]	.43**		
I(pb_med * pb_soc)	-0.04	[-0.10, 0.02]	-0.32	[-0.81, 0.17]	.01	[-.02, .04]		R2 = .378**	Delta R2 = .011
								95% CI [.21,.49]	95% CI [-.02, .04]

Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also significant.

b represents unstandardized regression weights. beta indicates the standardized regression weights.

sr2 represents the semi-partial correlation squared. r represents the zero-order correlation.

Square brackets are used to enclose the lower and upper limits of a confidence interval.

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

Table 8

Regression results using Drug consumption as the criterion

Predictor	b	<i>b</i>		<i>beta</i>		<i>sr<sup>2</sup></i>		r	Fit	Difference
		95% CI [LL, UL]	beta [LL, UL]	95% CI [LL, UL]	sr <sup>2</sup>	95% CI [LL, UL]	r			
(Intercept)	3.53**	[2.80, 4.26]								
Medical issues	0.32**	[0.20, 0.44]	0.43	[0.27, 0.60]	.18	[.06, .31]	.49**			
Social issues	0.28**	[0.15, 0.42]	0.36	[0.19, 0.52]	.12	[.02, .23]	.43**			
								R2 = .366**		
								95% CI [.21,.48]		
(Intercept)	3.02**	[1.95, 4.08]								
Medical issues	0.50**	[0.20, 0.80]	0.67	[0.27, 1.07]	.07	[-.01, .16]	.49**			
Social issues	0.41**	[0.18, 0.63]	0.51	[0.23, 0.79]	.08	[-.00, .17]	.43**			
Interaction	-0.04	[-0.10, 0.02]	-0.32	[-0.81, 0.17]	.01	[-.02, .04]		R2 = .378**	ΔR2 = .011	
								95% CI [.21,.49]	95% CI [-.02, .04]	

Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also significant. b represents unstandardized regression weights. beta indicates the standardized regression weights. sr<sup>2</sup> represents the semi-partial correlation squared. r represents the zero-order correlation. LL and UL indicate the lower and upper limits of a confidence interval, respectively.\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

**Situation 3 :** Nous souhaitons obtenir un tableau de l'**analyse de régression linéaire à deux modèles hiérarchiques** avec pour premier bloc la prédiction des explications complotistes ([ExplC](#)) par l'âge ([Age](#)), le genre ([Gender](#)) et l'orientation politique ([OrienPol](#)), et dans un second bloc la prédiction par la satisfaction dans la vie ([Sat](#)), la méfiance politique ([DtrustPol](#)), le sentiment de manque de contrôle ([LackCtrl](#)), l'autoritarisme de droite ([RWA](#)), la croyance en un moment dangereux ([BDW](#)) et l'adhésion aux théories du complot ([TC](#)), variables tirées de la base de données [Database](#).<sup>1</sup> Nous souhaitons un IC de 95%. Ce tableau est le 9<sup>ème</sup> de notre article.

<sup>1</sup> Cet exemple est tiré du chapitre 4.4.8.4 Méthode de sélection hiérarchique.

```

bloc01 <- lm(ExplC ~ Age + Gender + OrienPol,
  data = Database)
bloc02 <- lm(ExplC ~ Age + Gender + OrienPol +
  Sat + DtrustPol + LackCtrl + RWA + BDW + TC,
  data = Database)
apa.reg.table(bloc01, bloc02,
  table.number = 9,
  prop.var.conf.level = .95,
  filename = "LMhier-ExplC.doc")

```

Table 9

Regression results using ExplC as the criterion

Predictor	b	b_95%_CI	beta	beta_95%_CI	sr <sup>2</sup>	sr2_95%_CI	r	Fit	Difference
(Intercept)	3.14**	[1.63, 4.64]							
Age	0.04	[-0.02, 0.09]	0.18	[-0.08, 0.44]	.03	[-.06, .12]	.19		
Gender	-0.17	[-0.63, 0.28]	-0.10	[-0.37, 0.17]	.01	[-.04, .06]	-.12		
OrienPol	0.00	[-0.13, 0.14]	0.00	[-0.26, 0.27]	.00	[-.00, .00]	-.00		
								R2 = .046	
								95% CI[.00,.14]	
(Intercept)	1.55*	[0.07, 3.03]							
Age	0.03	[-0.01, 0.07]	0.17	[-0.04, 0.38]	.03	[-.03, .08]	.19		
Gender	-0.38*	[-0.75, -0.01]	-0.22	[-0.44, -0.01]	.04	[-.03, .11]	-.12		
OrienPol	0.05	[-0.08, 0.17]	0.09	[-0.16, 0.34]	.00	[-.02, .03]	-.00		
Satisfaction	-0.01	[-0.18, 0.15]	-0.02	[-0.26, 0.22]	.00	[-.01, .01]	-.08		
DistrustPol	0.34**	[0.13, 0.54]	0.45	[0.18, 0.73]	.11	[-.01, .22]	.59**		
LackControl	0.02	[-0.14, 0.19]	0.03	[-0.19, 0.25]	.00	[-.01, .01]	.19		
RWA	-0.19	[-0.45, 0.06]	-0.22	[-0.50, 0.07]	.02	[-.03, .07]	.05		
BDW	0.11	[-0.08, 0.30]	0.19	[-0.13, 0.51]	.01	[-.03, .06]	.41**		
TC	0.16*	[0.01, 0.31]	0.25	[0.02, 0.48]	.05	[-.03, .12]	.48**		
								R2 = .520** Delta R2 = .474**	
								95% CI[.22,.59] 95% CI[.29, .66]	

Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also significant.

b represents unstandardized regression weights. beta indicates the standardized regression weights.

sr<sup>2</sup> represents the semi-partial correlation squared. r represents the zero-order correlation.

Square brackets are used to enclose the lower and upper limits of a confidence interval.

\* indicates p &lt; .05. \*\* indicates p &lt; .01.

Table 9

Regression results using Conspiratorial explanation of economy as the criterion

Predictor	b	b		beta		sr <sup>2</sup>	r	Fit	Difference
		95% CI [LL, UL]	beta 95% CI [LL, UL]	95% CI [LL, UL]	sr <sup>2</sup> 95% CI [LL, UL]				
(Intercept)	3.14**	[1.63, 4.64]							
Age	0.04	[-0.02, 0.09]	0.18	[-0.08, 0.44]	.03	[-.06, .12]	.19		
Gender	-0.17	[-0.63, 0.28]	-0.10	[-0.37, 0.17]	.01	[-.04, .06]	-.12		
Policy Orientation	0.00	[-0.13, 0.14]	0.00	[-0.26, 0.27]	.00	[-.00, .00]	-.00		
								R2 = .046	
								95% CI[.00,.14]	
(Intercept)	1.55*	[0.07, 3.03]							
Age	0.03	[-0.01, 0.07]	0.17	[-0.04, 0.38]	.03	[-.03, .08]	.19		
Gender	-0.38*	[-0.75, -0.01]	-0.22	[-0.44, -0.01]	.04	[-.03, .11]	-.12		
Policy Orientation	0.05	[-0.08, 0.17]	0.09	[-0.16, 0.34]	.00	[-.02, .03]	-.00		
Satisfaction	-0.01	[-0.18, 0.15]	-0.02	[-0.26, 0.22]	.00	[-.01, .01]	-.08		
Policy Distrust	0.34**	[0.13, 0.54]	0.45	[0.18, 0.73]	.11	[-.01, .22]	.59**		
Lack of Control	0.02	[-0.14, 0.19]	0.03	[-0.19, 0.25]	.00	[-.01, .01]	.19		
RWA	-0.19	[-0.45, 0.06]	-0.22	[-0.50, 0.07]	.02	[-.03, .07]	.05		
BDW	0.11	[-0.08, 0.30]	0.19	[-0.13, 0.51]	.01	[-.03, .06]	.41**		
TC	0.16*	[0.01, 0.31]	0.25	[0.02, 0.48]	.05	[-.03, .12]	.48**		
								R2 = .520** ΔR <sup>2</sup> = .474**	
								95% CI[.22,.59] 95% CI[.29, .66]	

Note. A significant b-weight indicates the beta-weight and semi-partial correlation are also significant. b represents unstandardized regression weights. beta indicates the standardized regression weights. sr<sup>2</sup> represents the semi-partial correlation squared. r represents the zero-order correlation. LL and UL indicate the lower and upper limits of a confidence interval, respectively. RWA = Right-Wing Authoritarianism. BDW = Beliefs in a Dangerous World. TC = Theory of Conspiracy endorsement.

\* indicates p &lt; .05. \*\* indicates p &lt; .01.